Better confidence intervals in simulation-based inference

François Rousset[1], Raphaël Leblois[2], Arnaud Estoup[2], and Jean-Michel Marin[3]

[1]ISEM, Univ Montpellier, CNRS, IRD, 34095 Montpellier, France
[2]CBGP, INRAE, CIRAD, IRD, Institut Agro, Univ Montpellier, 34980 Montferrier-sur-Lez, France
[3]Institut Montpellierain Alexander Grothendieck, Univ Montpellier, 34095 Montpellier, France

October 1, 2024

Corresponding author:
F. Rousset, CC065 ISEM, Université de Montpellier, Pl. E. Bataillon 34095 Montpellier Cedex, France. +33467144630
francois.rousset@umontpellier.fr

Abstract

We describe and evaluate a method of statistical inference of model parameters, which revisits the idea of inferring a likelihood surface using simulation when the likelihood function cannot be evaluated. The method aims in particular to provide confidence intervals with controlled coverage, and its performance is assessed accordingly. It is based on a combining the random forest machine-learning method, and multivariate Gaussian mixture (MGM) models, in an effective inference workflow, here used to fit models with up to 15 variable parameters. Masked autoregressive flows, a deep learning technique, is also tested as an alternative do MGM models. The method is compared to that of approximate Bayesian computation (ABC) with random forests, with which it shares some technical features, on scenarios of inference of historical demography from population genetic data. These comparisons highlight the importance of an iterative workflow for exploring the parameter space efficiently. For equivalent simulation effort of the data-generating process, the new summary-likelihood method provides better control of coverage than ABC with random forests, and than generally reported for ABC methods.

Significance statement

Simulation-based methods of statistical inference are being actively developed in several fields. However, the accuracy of credibility or confidence intervals they produce is poorly characterised. Furthermore, while sequential methods of exploring the parameter space may be superior, they depend on various control parameters that need to be adjusted in each application, or are hampered by their own computational burden. As a result, non-sequential methods remain widely used. We present a new sequential workflow for inferring model parameters, which is fully automated and provides likelihood-ratio based confidence intervals. Comparisons with a related non-sequential method are carried out on demo-genetic inference scenarios, highlighting the increased accuracy of the results despite the limited amount of simulation that is required.

1 Introduction

The likelihood function is a classical component of efficient statistical inference methodologies. In instances where the likelihood function cannot be computed in a reasonable time frame, alternative statistical methods can be employed. These include moment-matching techniques when the analytical results for moments of a response variable are known, or approaches based on the simulation of the putative data-generating process. Among the latter simulation-based methods, approximate Bayesian computation (ABC) has been particularly developed, notably for applications in population genetics (Tavaré *et al.*, 1997; Beaumont *et al.*, 2002; Bertorelle *et al.*, 2010; Beaumont, 2010), but also in diverse other fields of biology (Schälte *et al.*, 2020) and beyond (e.g., Akeret *et al.*, 2015; Sisson *et al.*, 2019).

The idea of estimating the likelihood function by simulation is less developed, although it goes back at least to Diggle *et al.* (1984). This approach has had little follow-up in the form of effectively applied likelihood methods for more complex models, and for data summarized by many descriptive statistics ("summary statistics"). The "synthetic-likelihood" method (Wood, 2010) may be seen as derived from it, except that it assumes the summary statistics to have a multivariate normal distribution for each parameter value. An extension of Diggle *et al.*'s approach not making this assumption has been described by Rousset *et al.* (2017). As this is a likelihood inference using the information retained in the summary statistics, we refer to this method as "summary-likelihood (SL) inference".

The aforementioned methods all address the issue of inferring the parameters $\boldsymbol{\theta}$ of a hypothetical data-generating process, utilising simulations of this process for varying values of the vector $\boldsymbol{\theta}$. ABC methods are employed for the purpose of inferring posterior distributions for parameters, given a prior distribution for $\boldsymbol{\theta}$. Subsequently, credible intervals can be derived from the posterior distribution. As an alternative approach, likelihood-ratio based confidence intervals can be computed when a likelihood surface is inferred. A preliminary comparison of confidence intervals derived from the summary-likelihood method with credibility intervals obtained from an alternative ABC method (Peter *et al.*, 2010) was presented in Rousset *et al.* (2017) for a three-parameter model and using six summary statistics. The results of this comparison were consistent with the general theoretical expectation of a more uniform control of coverage by confidence intervals than by credibility intervals. It is anticipated that credibility intervals will control coverage on

average over the chosen prior distribution for parameters, whereas confidence intervals are defined to demonstrate a more uniform control of coverage over the entire parameter range.

In Rousset *et al.* (2017) as in Diggle *et al.* (1984) and Wood (2010), values of the parameter vector $\boldsymbol{\theta}$ are drawn and for each such value, the distribution of summary statistics is estimated using a moderately large number of simulations of the data-generating process. In this study, we present a more efficient summary-likelihood inference workflow that requires significantly fewer simulations to provide accurate inferences. This workflow enables the inference of more parameter-rich models when simulating observations is computationally intensive. The new workflow employs a reference table of simulations with a single simulated observation for each drawn value of $\boldsymbol{\theta}$. This approach aligns with the general methodology of ABC methods.

The idea of estimating the likelihood function for summary statistics from an ABC reference table already appears in the ABC literature (e.g., Fan *et al.*, 2013; Rubio *et al.*, 2013; Papamakarios *et al.*, 2019), but this has not yet led to accessible and validated software implementations of summary-likelihood inference. Indeed, a potentially substantial drawback of this idea is that inferring the likelihood surface in high dimensions is a complex task. Likewise, inferring a joint posterior distribution is difficult, but even when a full joint posterior distribution is inferred, practical inferences still typically only use one or two-dimensional marginal posterior distributions derived from the full joint one, so the importance of accurately estimating the full joint one is hard to appreciate. For inference of likelihoods or joint full posterior distributions, the importance of iterative (or "sequential") methods of construction of the reference table has been repeatedly stressed (e.g. Del Moral *et al.*, 2006; Toni *et al.*, 2009; Blum *et al.*, 2010; Cranmer *et al.*, 2019; Lueckmann *et al.*, 2021). Modern variants employ neural networks for density estimation (Blum *et al.*, 2010; Papamakarios *et al.*, 2016; Lueckmann *et al.*, 2017; Papamakarios *et al.*, 2019; Greenberg *et al.*, 2019). However, in practice, non-iterative ABC methods remain widely used (e.g., Beaumont *et al.*, 2002; Blum *et al.*, 2010; Pudlo *et al.*, 2016). As will be illustrated in this work, for a given simulation effort sufficient for good confidence interval inference by the iterative summary-likelihood method, reference tables built non-sequentially can be inappropriate to obtain good confidence or credibility intervals.

When the number of summary statistics is large, the inference of a likelihood surface is greatly facilitated by applying dimension-reduction methods to summary statistics, among which various machine-learning methods can

be used. The random-forest method of non-parametric regression will be used for this purpose here. This method is also used in ABC-RF (Pudlo *et al.*, 2016; Raynal *et al.*, 2019; Collin *et al.*, 2021). There are however notable differences between the summary-likelihood and ABC-RF methods, beyond the nature of the intervals sought, and whether the workflow is iterative or not. In particular, ABC-RF is currently constrained to infer marginal posterior distributions for each parameter separately. By contrast, the likelihood surface inherently retains information about the joint effect of parameters on the likelihood of the data. It is also possible to extend ABC-RF for the inference of multivariate parameters, using distributional random forests Dinh *et al.* (2024).

It would appear that a significant number of ABC methods are unable to effectively control the coverage of credibility intervals. In their study, Raynal *et al.* (2019) evaluated this coverage for ABC-RF, as well as for basic rejection ABC and its elaborations using adjusted local regression (Beaumont *et al.*, 2002), ridge regression (Blum *et al.*, 2013), or neural networks (Blum *et al.*, 2010). They found that ABC-RF intervals were conservative, with 100% coverage for 95% nominal probability. Additionally, they found that the coverage of other ABC methods was contingent upon a required rejection threshold, which is challenging to accurately set a priori. Consequently, they also generated anti-conservative intervals for certain threshold values. As observed by Hermans *et al.* (2022), the issue of coverage control is seldom addressed in the machine learning literature. The deep learning-based methods they examined were all found to produce anti-conservative intervals, indicating a clear need for improvement in the coverage of intervals in simulation-based inference methods.

In this paper, we present an evaluation of the performance of our new summary-likelihood workflow using simulation scenarios of inference of demographic history by analysis of population genetic data. A toy example completes the presentation. We show that the new workflow allows better control of the coverage of confidence intervals than previously reported for ABC methods. The main deviations from ideal control appear due to lack of information about parameters in the data, a common feature of demo-genetic inferences, and of other fields of application of simulation-based inference (e.g., Daly *et al.*, 2018; Fan *et al.*, 2019; Auger-Méthé *et al.*, 2021). We further investigate differences with ABC-RF, highlighting the importance of using an iterative workflow to explore the parameter space.

2 The inference workflow

Based on an initial estimation of the likelihood surface derived from a limited set of preliminary simulations, new parameter points are sampled with greater probability in regions of high likelihood, with the objective of more accurately inferring the likelihood surface in such regions. Such sampling should still permit further exploration to prevent entrapment in the initial high-likelihood region. These sampling steps are repeated iteratively to augment the reference table and to obtain progressively more precise inferences of the likelihood ratio surface. In this section, we initially present a simplified view of this iterative inference workflow, introducing fundamental terminology and notation, before describing potential modifications that may be necessary to improve performance.

2.1 Concept of instrumental distribution

The likelihood $L(\boldsymbol{\theta}; \mathcal{D})$ of a parameter vector $\boldsymbol{\theta}$ given observed data $\mathcal{D}$ is generally defined, up to a constant factor, as the probability or the probability density $\mathrm{P}_{Y;\boldsymbol{\Theta}}(\mathcal{D}; \boldsymbol{\theta})$ of the data $\mathcal{D}$, viewed as a function of the parameters for fixed data. Following well-established notation that we will repeatedly use below, the indices of density P denote the random variables whose values are the arguments of the function. Y values are realizations of an hypothetical data-generating process.

To estimate the likelihood function, we view the $\boldsymbol{\theta}$ vectors in the cumulative reference table as samples from an "instrumental" distribution. After a few iterations, this distribution is usually quite different from the distribution used to construct the initial reference table. In this respect, this distinction is similar to that between the prior and the "proposal" distribution in Papamakarios *et al.* (2019), but the instrumental distribution is generally distinct from the proposal distribution used in a given iteration for drawing parameter points.

Given an instrumental distribution with density $i_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$, we can write the likelihood as

$$L(\boldsymbol{\theta}; \mathcal{D}) = \mathrm{P}_{Y,\boldsymbol{\Theta}}(\mathcal{D}, \boldsymbol{\theta}) / i_{\boldsymbol{\Theta}}(\boldsymbol{\theta}), \tag{1}$$

in terms of the joint density $\mathrm{P}_{Y,\boldsymbol{\Theta}}(S, \boldsymbol{\theta}) = i_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \mathrm{P}_{Y;\boldsymbol{\Theta}}(S; \boldsymbol{\theta})$ of samples S and parameters $\boldsymbol{\theta}$. Accordingly, to estimate the likelihood function from the reference table, one can in principle first estimate a joint density $\mathrm{P}_{Y,\boldsymbol{\Theta}}$ of

samples and parameters, from their realized joint distribution in the reference table. One can then divide the value in $Y = \mathcal{D}$ of this estimated joint density function by an estimate of $i_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$ to obtain the likelihood function.

Any instrumental distribution with appropriate support can be used in the above computation, so any of the prior distributions considered in ABC can be used. However, the choice of the instrumental distribution should in practice be determined by its suitability for accurately estimating the likelihood function in parameter ranges of interest, which differs from additional requirements for prior distributions usually considered in the literature on Bayesian methods. In the present workflow, in order to preferentially fill the region of parameter space with a high probability, the instrumental distribution is updated iteratively, starting from a broad flat initial distribution.

For clarity of further exposition, we emphasize here the semantic distinction between the "samples" in the reference table and the "data" $\mathcal{D}$ to be analyzed. Even for simulated data, the data-generating parameters values $\boldsymbol{\theta}^{\dagger}$ are not an information to be used by the inference workflow. By contrast, the knowledge of sample-generating parameters in the reference table is essential, and this delineates the distinction between "data" and "samples". The same semantic logic could be used to distinguish the data-generating process from the sample-generating process in a study of the effect of model mis-specification.

2.2 From raw statistics to projected statistics

Here, as in most applications of ABC, the information provided to the inference method is typically a vector of statistics $\mathbf{u}(\mathcal{D})$, summarizing the observed data $\mathcal{D}$ which is the information available to the analyst (in some applications $\mathbf{u}(\mathcal{D})$ may just be a vectorial representation of the full $\mathcal{D}$). The simulated samples for each drawn $\boldsymbol{\theta}$ must then also be expressed as a vector of summary statistics $\mathbf{u}(S)$. As with ABC, it is up to the user to provide statistics that are informative for a given inference problem.

In our SL method, the potentially large number of *raw* summary statistics describing the data or the simulated samples in the reference table (up to 130 summary statistics in our simulations), is typically reduced by a *projection step* to a smaller number of *projected* summary statistics $\mathbf{p}[\mathbf{u}(D)]$, in order to reduce the dimension of the joint distribution, of parameters and of statistics, which is estimated. "Projection" here refers to the idea of non-linear projection to a lower-dimensional space. We obtain the projected summary

statistics by performing a non-parametric regression of each parameter θ_j on the raw summary statistics. This regression is performed using a variant of the random-forest method, combining features of its original version (Breiman, 2001) and of "extremely-randomized trees" (Geurts *et al.*, 2006). Various other projection methods could be used in SL, provided they avoid overfitting. The random-forest method was retained for the same reasons as in ABC-RF: it is fast, efficient, does not require ad-hoc adjustment of control settings for each application, and the out-of-bag predictions can be used to easily avoid overfitting (e.g., Hastie *et al.*, 2009). It thus fulfills the need for a method which can be applied automatically in any inference problem (Raynal *et al.*, 2019). This choice also facilitates comparison with ABC-RF as differences between the performance of ABC-RF and summary likelihood cannot be attributed to the choice of widely different projection methods.

In our use of random forests, a joint vector $\mathbf{p}[\mathbf{u}(S)]$ of projected summary statistics is defined, for each simulated sample S, as the vector of predictions $\hat{\theta}_j(S)$ of each element θ_j of the $\boldsymbol{\theta}$ parameter vector by its random-forest regression. This reduces the number of statistics to the number of parameters, which is the minimum required for identifiability of the parameters. Additional statistics can be retained insofar as computations do not become impractically slow, but have not been considered here.

An initial reference table is thus constructed as follows. Parameter vectors are drawn from the initial instrumental distribution $i^{(0)}_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$ for parameters, and one sample S (i.e., one realization of the biological and sampling process) is drawn for each parameter vector. Raw summary statistics $\mathbf{u}(S)$ are computed, and projected statistics $\mathbf{p}[\mathbf{u}(S)]$ are deduced from them for each such sample.

2.3 Estimating the summary-likelihood function

The likelihood function for projected statistics is written

$$L(\theta; \mathbf{p}[\mathbf{u}(\mathcal{D})]) = \mathrm{P}_{T,\boldsymbol{\Theta}}(\mathbf{p}[\mathbf{u}(\mathcal{D})], \boldsymbol{\theta})/i_{\boldsymbol{\Theta}}(\boldsymbol{\theta}), \tag{2}$$

in terms of the joint density $\mathrm{P}_{T,\boldsymbol{\Theta}}$ of projected statistics and parameters.

To model this joint density from the reference table, we use by default a Multivariate Gaussian Mixture (MGM) model. We have also used Masked Autoregressive Flows (MAFs, Papamakarios *et al.*, 2017), a deep-learning method to infer unconditional or conditional probability densities. The train-

ing of MAFs appears to be too slow to be evaluated in the same systematic way as the MGM workflow, but some comparisons are presented below.

Once a joint density estimate $\hat{\mathrm{P}}_{T,\boldsymbol{\Theta}}$ is obtained using MGM models, the likelihood for any $\boldsymbol{\theta}$ is estimated by

$$\hat{L}(\boldsymbol{\theta}; \mathbf{p}[\mathbf{u}(\mathcal{D})]) = \hat{\mathrm{P}}_{T,\boldsymbol{\Theta}}(\mathbf{p}[\mathbf{u}(\mathcal{D})], \boldsymbol{\theta})/\hat{\imath}_{\boldsymbol{\Theta}}(\boldsymbol{\theta}), \tag{3}$$

where, when the joint density estimation uses MGM models, the estimate $\hat{\imath}(\boldsymbol{\theta})$ of the instrumental density is easily deduced from the joint density estimate by marginalization of the latter density over the projected statistics. One may think that such estimation is not needed, at least in the first iteration when the instrumental distribution typically has some simple known form. However, in subsequent iterations such knowledge is not available because the instrumental density (as automatically generated by rules discussed later) is implicit and complex.

When MAFs are used, the likelihood function can be directly estimated by training a conditional MAF, in order to learn the conditional density of samples given any $\boldsymbol{\theta}$ values, on the joint distribution of parameters and statistics in the reference table.

Maximum summary-likelihood estimates (summary-MLEs or SMLEs) and summary-likelihood ratio tests (summary-LRTs) are deduced from the inferred likelihood surface, using numerical maximization.

2.4 Iterative refinements

Given a current estimate of the likelihood function, new parameter values are drawn, simulations of the process are performed and added to the reference table, and the above steps of the workflow are repeated on the incremented reference table. The sampling procedure allows exploration of the parameter space beyond the previously sampled ranges, and users can specify "absolute" bounds that should not be exceeded during such exploration. The software also allows users to specify not only ranges for each parameter, but also arbitrary additional constraints on any combination of parameters (this is used in some of our simulations).

The rules for drawing n_{i+1} new parameter vectors after iteration i are detailed in the Supplementary Information (Section S.1.1). They are defined to preferentially fill the region of parameters with a given minimum likelihood ratio $l_{\min}$ relative to the current summary-MLEs. In practice, $l_{\min}$ was set to

the 95% threshold for two-dimensional confidence regions. The sampling also allows exploration beyond the boundaries of this "top" region. The sampling step within the "top" aims to fill it uniformly rather than in proportion to the likelihood ratio.

The iterative workflow allows the projections to be re-computed. However, random-forest computations on large tables take time, so some elaborations have been implemented to shorten them. First, for large reference tables, they use a subset of the reference table, mostly defined from points identified as belonging to the top of the likelihood surface (see Supplementary Material for details). Second, projections are not re-computed when more than 90% of the selected points were already used in the previous computation of the projections. The easiest way to accelerate inferences in parameter-rich models without substantially compromising performance may be to reduce this threshold so that projections are re-computed less often.

2.5 Implementation

All the above-described steps have been implemented in an automated workflow in the `Infusion` R package, which calls the `ranger` R package (Wright *et al.*, 2017) for random-forest methods, and the `Rmixmod` R package (Lebret *et al.*, 2015) for MGM modeling. For MAF training, we wrote an R interface to the implementation of MAFs in the `zuko` Python package (Rozet *et al.*, 2023).

2.6 Control of inference workflow

Our workflow potentially depends on many control parameters, but we set default values for all of them, which were used for all simulations in this paper, unless mentioned otherwise. These values, as described here and further detailed in Supplementary Section S.1, are automatically selected by our implemented procedures unless users specifically request different values.

2.6.1 Distribution modelling

We fit a MGM model for the probability distribution of points $\mathbf{y}$ (here in joint parameters and projected statistics space), of the form

$$p(\mathbf{y}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \boldsymbol{\Sigma}_k), \tag{4}$$

where K is the number of components, π_k is the probability that a point $\mathbf{y}$ belongs to component k, and $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ is the multivariate Gaussian distribution with mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. For a given number of components, all parameters are jointly estimated by maximizing the likelihood without constraints on parameter space (especially without any enforced relationship between the different covariance matrices).

The number of components K must be selected. By application of the rules detailed in the Supplementary Material, for a given number n of samples in the reference table, K is effectively the largest number $\leq \lceil n^{0.31} \rceil$ such that the number P of parameters of the MGM model (a function of K and of the dimension of $\mathbf{y}$) satisfies $P \leq n/4$.

2.6.2 Size of reference table

For fitting n_p parameters to a given dataset, a final number of $1000(3n_\mathrm{p} - 1)$ sample simulations are run. This should be considered as a minimal size of the reference table to obtain reasonably good performance given reasonable computation time when inferences are performed on series of 200 or more simulated datasets. Ideally, iterations should be repeated until a certain degree of accuracy is reached for the inference of the likelihood surface. A bootstrap procedure based on resampling of the MGM distributions is implemented to estimate the uncertainty in the inference of the likelihood surface, as quantified by the mean square error of the likelihood ratio at the inferred confidence bounds. This allows automatic termination of the iterations when a certain precision is reached. However, in the present simulation-based study the aim is rather to quantify the precision of inferences for a given size of the reference table, comparable to the size of reference tables used in an ABC inference. The final size was therefore fixed regardless of the achieved precision. Further details on the number of points added in each iteration are given in section S.1.1 in the Supplementary Information.

3 Design of simulation study

The concept of confidence interval is based on the control of coverage whatever the data-generating parameter values $\boldsymbol{\theta}^\dagger$, rather than only on average over a prior distribution, so we assess coverage for given $\boldsymbol{\theta}^\dagger$.

It is well understood that for fixed prior distributions, credibility intervals may become asymptotically equivalent to confidence intervals for "large samples", i.e., when the information contained in the parameters increases indefinitely with sample size. Credibility intervals may then be seen as asymptotic approximations to confidence intervals. But this asymptotic equivalence may fail when (i) coverage is evaluated conditionally for a parameter at the boundary of the prior distribution and (ii) for some definition of posterior intervals. In particular, intervals based on central quantiles of the posterior distribution ("central posterior intervals") will never cover the $\boldsymbol{\theta}^\dagger$ value when it is at a boundary.

To avoid results that would be impacted by the latter expected failure, we use $\boldsymbol{\theta}^\dagger$ values not approaching the bounds of their prior range. However, this boundary effect should not be overlooked: (i) it may be widely ignored by practitioners (reports of performance focusing on good marginal coverage encouraging such ignorance), and (ii) it is a real concern in practice insofar as software (including ABC-RF) often report central posterior intervals, which leads to the exclusion of parameter values at range bounds from reported intervals even when there is actually not enough information in the data to actually reject such values.

3.1 Toy example

We first consider two variants of a 15-parameter toy example where the number of raw summary statistics is the number of parameters, so that projections are not needed and inferences are relatively fast (see Supplementary Table S.1 for timings). In this example, we estimate the covariance parameters of a 5-dimensional normal distribution. The data-generating values, and the parametrization of this model in terms of the elements of the Cholesky factor for the covariance matrix, are detailed in Supplementary Section S.2.1. The summary statistics are defined as the 15 distinct elements of the observed covariance matrix of 50 draws from a multivariate normal distribution of dimension 5 with the given covariance matrix.

3.2 Scenarios of demographic history of populations

Sample simulations for the two scenarios described below have been run with version 1.1.36 of the `diyabc` command-line simulation program (Collin *et al.*, 2021), with non-default `-o` argument allowing to provide the program with a table of sample-generating parameters. This allowed full control of the parameter values by the summary-likelihood procedures and avoided some boundary effects (see details in Supplementary Section S.4.1.1), which may otherwise occur when using `diyabc`.

3.2.1 Origin of invasive ladybird populations

This scenario (Figure 1, left) was already taken as an example for ABC-RF analyses by Pudlo *et al.* (2016). It is motivated by the invasion of the ladybird beetle *Harmonia axyridis* in Europe, and by a dataset of genotypes at 18 microsatellite loci, from a total of 126 individuals from four populations (Lombaert *et al.*, 2011). The fitted model has 8 parameters: effective population sizes N_1 to N_4, admixture time t_1, admixture proportion r_{a}, and two mutation parameters $\bar{\mu}$ and $\bar{p}$. We estimated the composite parameters $\log(N_{\mathrm{i}}\bar{\mu})$ instead of the N_is. Further details on model, data, and sampled parameter ranges are given in Supplementary Section S.3.1.

3.2.2 Human admixture scenario

For this second scenario (Figure 1, right), the model and data were also already considered in previous papers on ABC-RF (Raynal *et al.*, 2019; Collin *et al.*, 2021). It is a scenario of admixture between populations of European and African ancestry in America. More precisely, after an ancient demographic change at time t_4 in the ancestral African population, an out-of-Africa colonization event occurs at time t_3 that gives an ancestral out-of-Africa population which secondarily splits into one European population and one East Asian population at time t_2. Finally, a genetic admixture event occurs between populations of European and African ancestry in America at the time t_1, with a proportion r_{a} of European ancestry. Additional parameters (N_{bn3}, N_{bn4}, N_{bn34}, d_3, d_4, and d_{34}) describe the magnitude of bottleneck events associated to colonization events along the population tree. As detailed and justified in Supplementary Section S.4.1.1, we re-parametrized the model in terms of $t_{12} = t_2 - t_1$, $t_{23} = t_3 - t_2$, $t_{34} = t_4 - t_3$.

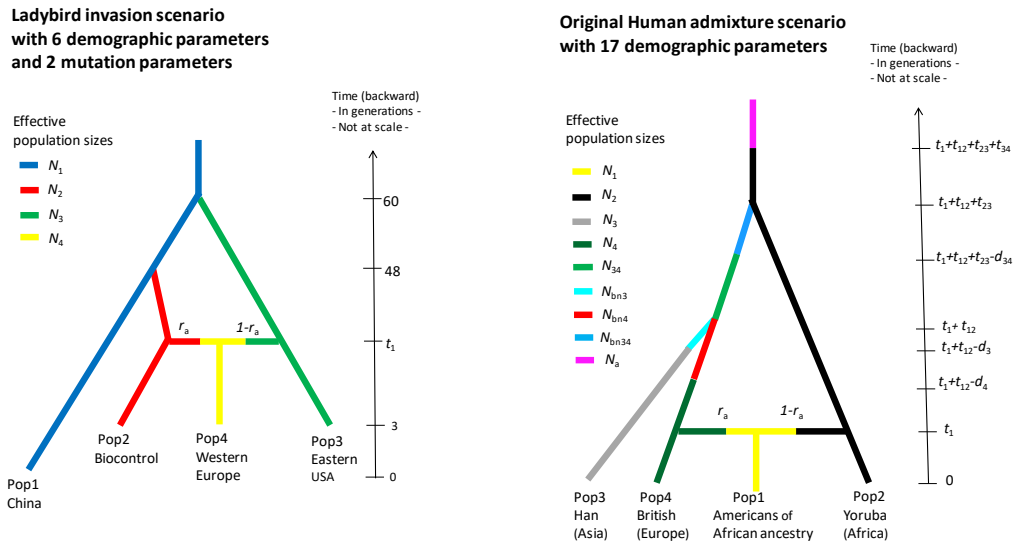

Figure 1: The two scenarios of historical demography.
Left: ladybird invasion scenario; right: Human admixture scenario.

We explored different versions of this model, with 7 or 13 estimated parameters, by assuming that the values of other parameters were fixed and known. A real dataset of 5000 SNP markers (extended to 10000 in some simulations), genotyped in four human populations (The 1000 Genomes Project Consortium, 2012), defines the sampling design of the simulated data. The four populations include Yoruba (Africa), Han (East Asia), British (Europe) and American individuals of African ancestry. Further details on model, data, and sampled parameter ranges are given in Supplementary Section S.4.1.1.

3.2.3 Choice of data-generating parameter values $\boldsymbol{\theta}^\dagger$

In principle, assessment of performance of confidence intervals calls for estimation of coverage for many different $\boldsymbol{\theta}^\dagger$ values. However, this would be unpractical here due to the high-dimensional parameter space and the computational cost of the simulations. Instead, we have used an automatic rule to select $\boldsymbol{\theta}^\dagger$ values in a relatively simple way that is both unaffected by our knowledge of the position of $\boldsymbol{\theta}^\dagger$ within the prior distributions used in ABC analyses, and more consistent with the real data than uniform sampling in the prior would be. In all cases, none of the $\boldsymbol{\theta}^\dagger$ values were in the lower 5%

or upper 5% of the prior ranges.

In practice, we first derived $\boldsymbol{\theta}^\dagger$ values from preliminary summary-likelihood fits of actual data, with one adjustment. In the Human admixture scenario, we set the admixture time t_1 to 6 generations, ignoring its often unrealistically high (> 15 generations) estimate from actual data (the simulations indeed show that this parameter cannot be estimated with precision), and other $\boldsymbol{\theta}^\dagger$ values were set to the constrained summary-MLEs given $t_1 = 6$.

We have departed from this automatic rule when seeking to find conditions where parameters could be estimated with more precision, and to answer questions raised by a first set of simulation results. This approach is sufficient to facilitate an exploration and comparison of the performance of SL and ABC-RF across a range of conditions.

Few parameters may be practically identifiable in the Human admixture scenario. For this reason, Raynal *et al.* (2019) and Collin *et al.* (2021) reported estimation performance for only a few parameters, with good performance only for the admixture rate r_{a} and for some composite parameters defined as ratios of population size to time interval parameters. However, such composite parameters complicate the interpretation of the results. For example, the parameter space is no longer defined only by box constraints (i.e., by the range of each parameter), and marginal prior distributions are no longer uniform. This implies that low relative RMSEs, which are defined relative to the marginal range of each parameter, are not necessarily indicative that the parameter is practically identifiable.

For these reasons, we considered composite parameters only in one of two considered 13-parameter variants of the Human admixture scenario. More specifically, in this variant we considered the following composite parameters, sometimes referred to as "bottleneck intensity" parameters in the colonization/invasion literature: $b_3 = d_3/N_{\mathrm{bn3}}$, $b_4 = d_4/N_{\mathrm{bn4}}$, and $b_{34} = d_{34}/N_{\mathrm{bn34}}$. Their possible values, in combination with other parameters, are subject to logical constraints with respect to the times of events in the demographic scenario, which can be represented as follows

$$\begin{gathered} d_3 = \lfloor b_3 N_{\mathrm{bn3}} \rfloor \geq 0, \qquad d_4 = \lfloor b_4 N_{\mathrm{bn4}} \rfloor \geq 0, \qquad d_{34} = \lfloor b_4 N_{\mathrm{bn4}} \rfloor \geq 0, \\ d_3 \leq t_2, \qquad d_{34} \leq t_{23}, \qquad d_4 \leq t_{12}, \text{ and } t_3 \leq t_4. \end{gathered} \tag{5}$$

Estimating additional population size parameters specific to the branches of the tree that are affected by bottlenecks is difficult. In this simulation variant, we therefore do not try to estimate two such parameters, N_3 and N_4.

Supplementary Table S.16 presents results for another 13-parameter variant without any composite parameters and with estimated N_3 and N_4.

The $\boldsymbol{\theta}^\dagger$ values used for each series of simulations are shown in Supplementary Table S.3 and Section S.4.1.2, respectively, for the ladybird invasion scenario and the Human admixture scenario.

3.3 Performance summaries

We evaluated the bias and root-mean-square error (RMSE) of the following point estimators: the summary-MLEs, and the mean and median of the posterior distributions for ABC-RF. Each Figure and Table for given data-generating parameter values is based on running the inference workflow on 400 simulated datasets (but only 200 datasets for some of the Supplementary Tables and Figures).

Parameter transformations were applied, mainly in order to homogenize the RMSEs of the estimators of transformed parameters in the population genetic scenarios. In practice this means that population size and mutation rate parameters were log-transformed, but for simplicity we also applied the following automatic transformation rule to other parameters based on their explored ranges: if upper bound is ≥ 500, log(1+.) or log(.) transformation is used, depending on whether the lower bound is zero or not. All logarithms are decimal. ABC-RF inferences used uniform priors on the transformed scales.

The RMSE summaries may not lead to clear conclusions. Which of ML and posterior estimates have lower RMSE depends on the location of data-generating values $\boldsymbol{\theta}^\dagger$ relative to the prior distribution used in a Bayesian inference (e.g., Casella *et al.*, 2002, p. 333), so we do not expect RMSE criteria to systematically favor one class of estimators over the other when examined conditionally given $\boldsymbol{\theta}^\dagger$ values. Moreover, an efficient posterior estimate is expected to have lower RMSE on average when $\boldsymbol{\theta}^\dagger$ values are chosen randomly in their prior distributions. Yet, systematic departures from such theoretical expectations can occur if one of the methods does not locate well its point estimates, as a result of poorly inferring the likelihood function or the posterior distribution.

Bias and RMSE of estimators will be reported on a scale relative to the width of the explored ranges. This is useful in particular to compare the RMSE of a summary-likelihood estimator to that of a non-identifiable parameter whose estimator would be uniformly distributed, and would then

have a relative RMSE equal to $1/\sqrt{12} \approx 0.289$. Another useful diagnostic pattern for poorly identifiable parameters is the relative variance of summary-MLE versus posterior estimates: the posterior mean estimator should have low variance, as it should approach the prior mean for all datasets. Thus, although an efficient posterior estimate is expected to have slightly lower RMSE on average over a prior distribution, a markedly lower ratio of the variance of posterior estimates to the variance of summary-MLEs instead suggests that the parameter is not well estimated by ABC-RF.

The actual coverage of nominal 95% intervals will be reported for both the confidence and credibility intervals. The full distribution of p-values of the (summary-)LRTs, which should be uniform in ideal conditions, is a more informative summary (not depending on a conventional level such as 95%), and will be reported in Figures. We also evaluate the performance of intervals provided by parametric bootstrap simulations, where bootstrap samples are drawn from the inferred distribution of projected summary statistics as function of the parameters (see Supplementary Section S.1.8 for details about this procedure). Various definitions exist for bootstrap-based confidence intervals (e.g., Davison *et al.*, 1997, Chapter 5). We thus compare the coverage of intervals defined by inverting profile-LRTs whose p-values are read from the χ^2 distribution, to two bootstrap intervals. One interval is defined by inverting the profile-LRTs whose p-values are read from the bootstrap distribution of the likelihood-ratio statistic. The second interval uses a Bartlett correction (Bartlett, 1937): it is defined by inverting p-values read from the χ^2 distribution, for the profile likelihood ratio statistic corrected by the estimated mean of this statistic in the bootstrap samples. These three measures of coverage are reported in the Tables as "profLR", "bootLR" and "BcorCI" respectively. We also considered percentile bootstrap intervals but these appear to have less predictable coverage over the different simulations conditions.

The reported coverage of credibility intervals is that of the central posterior intervals computed by the `abcrf` R package (Marin *et al.*, 2022). In contrast to them, highest posterior density (HPD) intervals can contain parameter values at the boundary, so we also implemented them from ABC-RF output. Because their coverage appears only marginally better than that of central intervals, they are reported only in the Supplementary Information.

It is not always reasonable to require exact control of the coverage or distribution of p-values. If there is no information about a parameter in the data (i.e., if likelihood is flat with respect to it), the conclusion of the analysis should be that there is no information. This lack of information

is represented by unbounded intervals with 100% coverage. On the other hand, weak information will result in intervals with more than the nominal coverage and in distributions of summary-LRTs that show a deficit of low p-values relative to the uniform distribution. These patterns readily occur in our simulations of demo-genetic scenarios.

4 Results

4.1 15-parameter toy model

The ability of the method to produce valid intervals in relatively high-dimensional problems can be checked by considering the estimation of the covariance matrix of a 5-dimensional multivariate normal distribution. This model has two variants, one where all 15 parameters are identifiable, the other where only 5 variances are identifiable. As this example mainly serves as a first check of the inference workflow in a case with a relatively high number of parameters and where we know which parameters are identifiable and which are not, no variation of data-generating parameters nor any comparison with ABC are presented.

Performance summaries are detailed in Supplementary Table S.2 and distributions of p-values of summary-LRTs are shown in Fig. 2. In each case 400 datasets were simulated and reference tables of 44000 samples were constructed independently for each dataset. Coverage is approximately controlled for identifiable parameters, mean coverage over the twenty rows of Table S.2 being 0.954 for the profile-likelihood ratio intervals and 0.96 for the Bartlett-corrected bootstrap intervals. There is expectedly a strong deficiency of low p-values for unidentifiable ones. Ideally, in the latter case, the distribution of p-values should be a step function in $p = 1$, and the observed non-stepwise distributions are the result of the fluctuations of the estimated surface compared to the exact one, these fluctuations being too small to result in low p-values. Overall, when parameters are not identifiable, the confidence intervals still have at least nominal coverage.

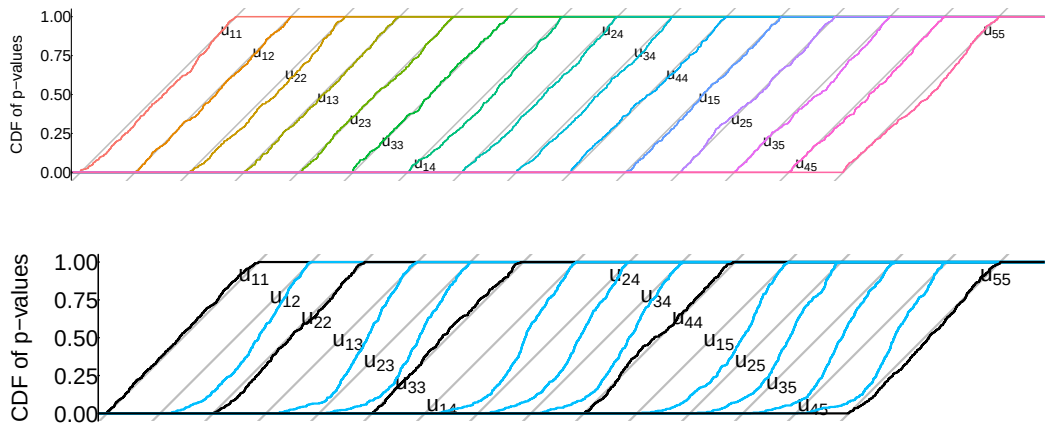

Figure 2: Distributions of p-values of summary-LRTs in multivariate-normal toy example
Distributions are shown for each parameter of the covariance matrix (i.e., each element u_{ij} of its Cholesky factor, eq. S.1 in Supplementary Material). Top: fully-identifiable model. Bottom: partially-identifiable model, with distributions for identifiable and non-identifiable parameters shown in black and blue, respectively.

4.2 Ladybird invasion scenario (8 parameters)

The simulation results reported in Table 1 show that there is little information about admixture rate r_{a}. Posterior estimators for this parameter appear to have low bias only because the data-generating value of r_{a} is close to the mean value of the prior distribution. This also explains why these posterior estimators have lower (conditional) RMSE than the summary-MLE. There is also weak information about $\log(N_4\bar{\mu})$, t_1, and $\bar{p}$. Accordingly, we found deficiencies of low p-values for those parameters. Intriguingly, such a pattern disappears in the bootstrap-corrected versions of the summary-LRTs (Table 1, "bootLR" and "BcorCI" columns, and Supplementary Fig. S.1, bottom). For the other, better estimated parameters (i.e. $\log(N_1\bar{\mu})$, $\log(N_2\bar{\mu})$, $\log(N_3\bar{\mu})$, and $\bar{\mu}$), the profile-LRTs also exhibit deficiencies of p-values (better seen in the Table than in the Figure). In these cases, the bootstrap corrections appear effective, with the Bartlett-corrected intervals having 95.4% coverage on

	Relative bias			Relative RMSE			coverage			
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI
$\log(N_1\bar{\mu})$	**0.000855**	0.01	0.00869	**0.0357**	0.0366	0.0367	0.97	0.965	0.968	**0.96**
$\log(N_2\bar{\mu})$	**-0.000489**	-0.0787	-0.0644	**0.0921**	0.148	0.15	0.975	**0.95**	0.945	0.98
$\log(N_3\bar{\mu})$	**0.00921**	-0.0409	-0.0246	**0.0961**	0.129	0.134	0.965	**0.952**	0.96	0.99
$\log(N_4\bar{\mu})$	0.0448	**0.0262**	0.056	0.264	**0.0508**	0.074	0.992	**0.952**	0.96	1
t_1	**0.0498**	0.128	0.0816	0.281	0.15	**0.132**	0.988	0.94	**0.948**	1
r_{a}	0.0172	0.0136	**0.00871**	0.171	**0.103**	0.121	0.972	0.96	**0.958**	1
$\log(\bar{\mu})$	**-0.0126**	-0.101	-0.0786	**0.0649**	0.129	0.111	0.972	**0.952**	0.945	0.985
$\bar{p}$	**0.00665**	0.0232	0.0183	0.172	**0.143**	0.146	0.968	0.952	**0.948**	0.988

Table 1: Performance summaries for the ladybird invasion scenario.
Bias and root-mean-square error (RMSE) are reported for the maximum-summary likelihood estimates (MSLE), and the posterior mean (postEv) and posterior median (postMed) for ABC-RF. Coverage of confidence intervals with nominal 95% level is reported for summary-likelihood inference (profLR) as implied by the distribution of p-values shown in Fig.S.1, for two forms of bootstrap correction described in the Text (profLR and BcorCI), and for the central intervals provided by ABC-RF (postCI). Bold font is used to emphasize for each parameter the bias value minimal in absolute value, the minimum RMSE, and the coverage closest to 0.95.

average.

The ABC-RF estimators of the different parameters often have higher bias, but there is no clear trend for RMSEs, as expected from basic theory. While the dependence of relative RMSE performance on location of the data-generating parameters in the prior distribution may explain some of the heterogeneity in apparent performance of the two methods, additional simulations show that more obscure specificities of the ABC-RF method contribute to this heterogeneity. Notably, the ABC-RF estimates for $\log(N_4\bar{\mu})$ are biased away from the prior mean (-0.5), their correlation with summary-ML estimates is low (0.215), and they have a much lower variance than the latter. Together, these results are not expected from general theory for posterior estimators, given the uniform priors and the fact that summary-MLE estimates do not exhibit a similar bias. Varying the $\log(N_4\bar{\mu})$ value, with other parameters being fixed, shows that the ABC-RF $\log(N_4\bar{\mu})$ estimator is biased away from the prior mean over much of the prior range (Supplementary Fig. S.2), whereas a bias toward the prior mean would be more commonly expected (e.g., Casella *et al.*, 2002, Section 7.3.4). Yet, it has lower averaged RMSE (Supplementary Table S.4) than the summary-ML estimator, as expected when comparing the prior-averaged performance of ML and posterior-mean estimators.

In Section S.3.2 of the Supplementary Information, we assessed performance of inferences for alternative data-generating parameter values derived from a preliminary fit of the actual data. These simulations exhibit even less information about parameters t_1 and r_{a}, and repeat the striking patterns of Fig. S.2 for estimation of $\log(N_4\bar{\mu})$.

Overall, this means that ABC-RF estimators can be strongly biased toward a value quite distinct from the prior mean, and their variance around this value may be small. ABC-RF estimates may then exhibit much higher or much lower RMSE than summary-MLEs, depending on the data-generating parameter values $\boldsymbol{\theta}^{\dagger}$ chosen, and this pattern cannot be simply interpreted in terms of location of $\boldsymbol{\theta}^{\dagger}$ relative to the prior mean.

4.3 7-parameter human admixture scenario

This inference scenario is obtained by fixing the following ten parameters from the full 17-parameter scenario schematized in the right part of Fig. 1 : $\log(N_1) = 4.1$, $\log(N_3) = 4.3$, $\log(N_4) = 3.5$, $r_{\mathrm{a}} = 0.2$, $d_3 = 42$, $N_{\mathrm{bn3}} = 160$, $d_4 = 9$, $N_{\mathrm{bn4}} = 98$, $\log(N_{34}) = 3.1$, $d_{34} = 24$. As inferences are relatively fast

in this scenario, it will be used to compare alternative procedures for parameter sampling, and for inference of the likelihood surface from the reference table, to the procedures used by default by the automated workflow.

4.3.1 Performance with default procedures

Performance summaries are reported in Table 2 and Supplementary Fig. S.4. The summary-likelihood method generally exhibits lower bias and RMSE than ABC-RF (Table 2). ABC-RF estimates can be strongly biased. In particular, we observe for parameter t_{23} a bias away from the prior mean, similarly to the pattern previously discussed for $\log(N_4)$ in the ladybird invasion scenario. In a transformed log scale, the data-generating value was $\log(1+t_{23}) = 1.5$, the mean summary-MLE was 1.66, the realized prior mean in the ABC-RF reference table was 1.855 and the mean ABC-RF estimate was 2.491. Examination of the log-likelihood values $\ell(\boldsymbol{\theta})$ for parameters values $\boldsymbol{\theta}$ from both the ABC-RF reference table and from the summary-likelihood reference table for a given dataset suggests an explanation, which we illustrate in Fig. 3, left panel (similar results have been obtained for other simulated datasets). All $\ell(\boldsymbol{\theta})$ are here the values of the log-likelihood function estimated by the summary-likelihood inference for the dataset. The points from the ABC-RF reference table form a cloud with a likelihood maximum near $x = 2.5$. This readily explains the mean posterior estimate near 2.5. In this case, the uniform prior sampling used by the ABC-RF method hence appears to miss the more relevant parameter region. One consequence is that the coverage of posterior intervals for t_{23} is quite low (15.25%). On the other hand, the iterative exploration of the parameter space by the summary-likelihood workflow allows higher likelihood values to be attained for lower t_{23} values.

For both t_1 and N_{bn34}, estimates provided by ABC-RF are largely driven by the mean of the prior distribution, which is more expected in the case of a parameter with less apparent information about it (t_1) than with more information about it (N_{bn34}). The latter case might be explained as a less striking effect of the failure of the ABC-RF analysis to explore the narrow region of parameter space with high likelihood (Fig. 3, right panel). Two consequences of this failure, combined with attraction of posterior estimates towards the prior mean are that the coverage of "central" posterior intervals for N_{bn34} is distinctively low (88.5%), but that HPD intervals are less affected (see Supplementary Table S.9).

Deficiencies of low p-values are observed for LRTs, without bootstrap

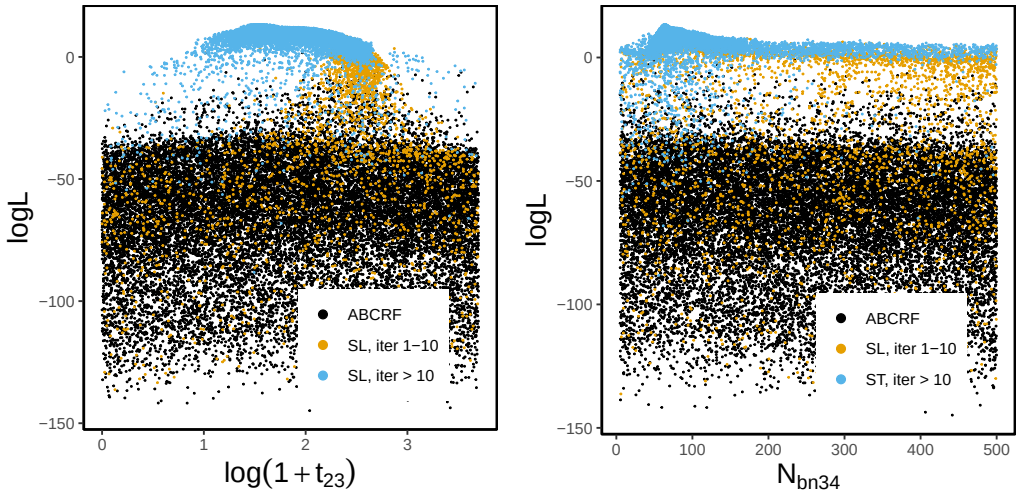

Figure 3: log-likelihoods of points $\boldsymbol{\theta}$ from ABC-RF and summary likelihood reference tables.
x-axis values are those of $\log(1 + t_{23})$ (left) or N_{bn34} (right) from each $\boldsymbol{\theta}$, for points from the ABC-RF reference table (in black) and for points from the summary inference reference table (in orange for points from the first 10 iterations, and in blue for points from later iterations). y-axis values are log-likelihood values according to the likelihood surface inferred in the final iteration of the summary likelihood workflow.

correction, for admixture time t_1, bottleneck size N_{bn34}, and time t_{23}, with corresponding higher than nominal coverage of the intervals. For t_1 there is little statistical information in the data: likelihood profiles are rather flat, and many estimates are at the lower bound of the parameter space. For N_{bn34} there is more information, as shown in particular by the low RMSE of summary-MLEs. The coverage of the Bartlett-corrected intervals for the other parameters is 94.6% on average.

To mitigate the effects of a low level of information in the data for some parameters, we performed additional simulations with datasets of 10,000 SNP rather than 5,000 as previously assumed. The detailed results reported in Supplementary Table S.9 show reductions of RMSE of summary-MLEs by a factor between 1.36 and 1.58 for the different parameters except N_{bn34}, which exhibits a reduction by 2.02. By contrast, the RMSE of the posterior mean estimator is reduced by a lower factor (between 0.93 and 1.28). Hence, the

	Relative bias			Relative RMSE			coverage			
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI
$\log(N_2)$	**0.00736**	0.101	0.106	**0.0478**	0.104	0.109	0.955	**0.948**	0.948	0.952
t_1	**0.101**	0.305	0.294	0.318	0.313	**0.309**	0.985	0.968	**0.965**	1
$\log(t_{12})$	**-0.00482**	-0.0338	-0.0298	**0.0463**	0.0563	0.0578	0.96	**0.942**	0.928	1
$\log(1+t_{23})$	**0.0392**	0.246	0.276	**0.07**	0.246	0.276	0.972	0.97	**0.95**	0.152
N_{bn34}	**0.00764**	0.36	0.326	**0.0194**	0.363	0.334	0.992	**0.98**	0.982	0.885
$\log(1+t_{34})$	0.00694	-0.0373	**-0.00496**	0.028	0.0468	**0.0229**	**0.948**	0.938	0.94	1
$\log(N_{\mathrm{a}})$	**0.00463**	0.0806	0.104	**0.0526**	0.0854	0.109	0.962	0.945	**0.948**	1

Table 2:
Performance summaries for the 7-parameter human admixture scenario. See Table 1 for definitions of the different performance criteria.

summary-likelihood results exhibit reductions in RMSE roughly as expected from doubling the information in the data (i.e. a reduction of RMSE by a $\sqrt{2}$ factor), but ABC-RF does not. This could reflect another benefit of a more efficient exploration of the region of interest of the parameter space by the iterative workflow.

4.3.2 Masked autoregressive flows

The final reference tables built for each dataset by the previous analyses using MGM models were reused to infer the likelihood surfaces using MAFs. The training of MAFs depends on various control parameters. Supplementary Table S.10 presents results for two sets of control parameters detailed in Supplementary Section S.1.7. The "PSM19" set of controls is the one recommended by Papamakarios *et al.* (2019, Section 5.1). It provides estimates with uniformly much larger biases and RMSEs than reported in Table 2 for MGM-based estimates for the same data and same reference tables. The "alternative" set of controls provides estimates with better biases and RMSEs, but still higher RMSEs than that of MGM-based estimates. On the other hand, coverage at the 95% level seems better controlled with the first set of controls than with the second. However, this comes with large overshoots of intermediate p-values, notably for $\log(1 + t_{23})$ and N_{bn34} (Supplementary Fig. S.5).

4.3.3 A synthetic reference table for all datasets

The fit of the simulated joint distribution for parameters and summary statistics in a reference table allows, in principle, the likelihood to be evaluated following eq. 3 for any dataset simulated under the same model as the reference table. Thus, a single such fit could in principle be used to perform inference for different real or simulated datasets $\mathcal{D}$. However, in practice, attempts to reuse reference tables built for other datasets may not be successful, if the likelihood evaluation is precise only after several iterations where the sampling of new parameter points is driven by the analyzed dataset. In order to check whether this occurs, we built a single synthetic reference table of the same size as the dataset-specific reference tables, by drawing 100 samples randomly from each of 200 such tables. We then re-inferred likelihood surfaces and intervals for the 200 datasets using this synthetic table. We found that the resulting intervals have poor coverage (Supplementary Ta-

ble S.11 and Fig. S.6). Thus, the inference workflow is best run essentially independently for each new dataset.

4.4 13-parameter Human admixture scenario

Results for the 13-parameter simulation scenario are presented in Table 3 and Supplementary Fig. S.7, with composite bottleneck intensity parameters defined as ratios of bottleneck duration to bottleneck population size: $b_3 = d_3/N_{\mathrm{bn3}}$, $b_4 = d_4/N_{\mathrm{bn4}}$, and $b_{34} = d_{34}/N_{\mathrm{bn34}}$. Summary-MLEs have lower absolute bias than posterior estimates, while for RMSEs the pattern is more heterogeneous, and can be interpreted as resulting from a combination of the effects previously considered to explain discrepancies between the two methods. A more detailed analysis for each parameter is presented in Supplementary Section S.4.3.2. In particular, it suggests that poor exploration of parameter space again affects estimation of $\log(1 + t_{23})$, but also of $\log(t_{12})$, N_{bn3}, $\log(N_{\mathrm{bn4}})$, and b_{34}. The admixture rate r_{a} and the composite bottleneck intensity parameters b_3 and b_4 appear well estimated by both methods, and population size N_2 may also be relatively well estimated by SL.

Other parameters are poorly estimated by both methods. For $\log(N_1)$ and N_{bn34} in particular, the position of the prior mean relative to the data-generating $\theta^\dagger$ value, rather than efficient use of information, explains the low RMSE of the ABC-RF estimates. In general, poor estimation of some parameters can be interpreted as demonstrating that there is weak information about them in the data, or that the estimation method fails to capture such information. To test the possibility that not enough iterations of the summary-likelihood workflow have been run to fill the top region of the likelihood surface, we increased the size of the reference tables from 38000 to 50000 simulated samples (i.e., 24 additional iterations), for the first 200 simulated datasets (Supplementary Table S.14). The effect of these additional iterations is, if any, too small to be detectable.

Intervals generally have higher than nominal coverage, which is not unexpected given the limited information present in the data about most parameters, and given additional boundary effects such as those detailed for b_3 in Supplementary Section S.4.3.4. The coverage approaches the nominal 95% for the best estimated parameters, and the bootstrap corrections partially correct the coverage for the other parameters. These results may be seen as evidence that the summary-likelihood method is able to provide confidence intervals for clearly identifiable parameters, from limited simulation effort in

such a 13-parameter scenario.

5 Discussion

In this study, we present and assess the performance of an automated workflow for summary-likelihood, a method of simulation-based inference based on inferring a likelihood surface for summaries of the data. The efficiency of the method is contingent upon its iterative procedure of exploration of the parameter space. In a 15-parameter toy example, the coverage of confidence intervals provided by summary-likelihood is nearly optimal. However, in cases where some parameters are not practically identifiable, the results are inherently more complex. In these cases, the intervals derived from the likelihood profiles for parameters with low information content are, as expected, too large. Nevertheless, bootstrap procedures appear to be a viable solution for obtaining better coverage.

Beyond documenting the general performance of the summary-likelihood method, our simulations provide a comparison with the ABC-RF method. Raynal *et al.* (2019) compared their ABC-RF method to some iterative (or "sequential") ABC workflows (ABC-PMC, Beaumont *et al.*, 2009; Prangle, 2017; SMC-ABC, Del Moral *et al.*, 2012). For a toy example, they found ABC-RF to better infer marginal posterior distributions, without the computational overhead of the iterative methods. However, they did not extend their comparisons to their population genetics example. By contrast, our results highlight the benefits of an iterative workflow for the exploration of parameter space, and are thus consistent with the premises of sequential ABC methods.

The comparisons with ABC-RF are based on previously considered simulation scenarios. Our findings indicate that ABC-RF is capable of producing estimates with favorable bias and RMSE characteristics, comparable to those of summary-likelihood. However, we also identified instances where its estimation performance diverged significantly, which can be attributed, at least in part, to the imperfect exploration of parameter space by non-iterative methods. Further evidence of this imperfect exploration can be observed in the markedly smaller reduction in the root mean square error (RMSE) of ABC-RF estimates relative to SL estimates when the amount of data is increased. This phenomenon was observed in the 7-parameter human admixture scenario.

	Relative bias			Relative RMSE			coverage			
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI
$\log(N_1)$	**-0.122**	-0.297	-0.28	0.364	0.302	**0.291**	0.968	**0.948**	0.955	1
$\log(N_2)$	**-0.00697**	0.0275	0.0236	0.0334	0.0341	**0.031**	0.962	0.935	**0.938**	1
t_1	**0.0764**	0.252	0.196	0.249	0.262	**0.221**	0.98	0.932	**0.94**	1
r_{a}	**-0.00155**	-0.00276	-0.0024	**0.0151**	0.0177	0.0192	**0.955**	0.925	0.93	0.998
$\log(t_{12})$	**-0.0209**	-0.0559	-0.0614	**0.0565**	0.0665	0.0777	0.998	**0.992**	0.995	1
d_3/N_{bn3}	**0.00805**	0.0308	0.0176	0.033	0.0392	**0.0279**	0.99	**0.968**	0.968	1
$\log(N_{\mathrm{bn3}})$	**-0.0757**	-0.133	-0.147	0.165	**0.138**	0.152	1	**0.972**	0.982	1
d_4/N_{bn4}	**0.00147**	0.0085	-0.00238	0.0252	**0.0242**	0.0262	0.985	**0.962**	0.972	1
$\log(N_{\mathrm{bn4}})$	**-0.0218**	-0.0691	-0.0865	0.185	**0.0812**	0.11	0.992	**0.965**	0.975	1
$\log(1+t_{23})$	**0.022**	0.205	0.235	**0.106**	0.207	0.237	0.985	**0.968**	0.97	1
d_{34}/N_{bn34}	**-0.0338**	-0.109	-0.108	**0.0551**	0.112	0.111	0.982	**0.972**	0.982	1
$\log(N_{\mathrm{bn34}})$	**0.0023**	0.0173	-0.0352	0.121	**0.0296**	0.0451	0.985	**0.96**	0.968	1
$\log(N_{\mathrm{a}})$	**-0.00994**	-0.0372	-0.0189	0.0421	0.0457	**0.0389**	0.965	**0.94**	0.935	1

Table 3: Performance summaries for the 13-parameter human admixture scenario
See Table 1 for definitions of the different performance criteria.

Moreover, we confirmed the previous observation that ABC-RF often produces 95% credibility intervals with 100% coverage. This is true even for the most easily estimated parameter. For instance, Raynal *et al.* (2019, Fig. 2) already found that 95% credibility intervals for the admixture rate r_{a} in the Human admixture scenario had 100% coverage. Comparison of credibility and confidence intervals shows that in such cases the credibility intervals consistently extend in both directions beyond the confidence intervals.

A further issue is that ABC-RF estimates may be significantly biased. This phenomenon can be attributed to the fact that sampling from prespecified priors, a common practice in non-sequential ABC methods, is inadequate for fully exploring the parameter space in certain models. Consequently, the credibility interval coverage provided by ABC-RF may be considerably diminished in such instances.

The summary-likelihood method was initially conceived as a way of inferring "the" likelihood surface, but it remains imperfect in several ways from that point of view. It can only provide a likelihood surface for the information retained in the projected summary statistics. Practical reasons for using random-forest methods have been recalled above (see Introduction, and Section From raw statistics to projected statistics). Nevertheless, different runs of the random-forest algorithm on the same table of "raw" summary statistics provide distinct predictors ("projections"). In the present workflow, the projections are updated as new samples are added to the reference table, contributing to variation of point estimates and of intervals over successive iterations. Increasing the number of decision trees in the random forest algorithm may reduce such variation and may be feasible in some iterations and/or some datasets, but was not practically feasible in the simulations studies of performance due to both time and memory constraints, as the "forest" of decision trees typically occupies large amounts of memory. Using forests of more than 2000 trees is indeed generally avoided (e.g., see Discussion in the Appendix of Collin *et al.*, 2021).

We found that replacing reference tables, each constructed for a different simulated dataset, with a single reference table of the same size leads to poor inference, even if it is obtained by subsampling the reference tables constructed for each different simulated dataset. While this result has little impact on the analysis of real datasets, each requiring a new reference table matching the details of the sampling design of the dataset, it prevents a drastic reduction of computation time in our simulation studies, which would be possible if a single reference table could be used for accurate inference from

all simulated datasets.

Further iterative ABC methods have been developed, notably sequential neural likelihood estimation (SNLE, Papamakarios *et al.*, 2019) and related workflows based on deep-learning methods that learn probability distributions and related functions (e.g., Sharrock *et al.*, 2024 and references therein). In the present work, we have also compared the default method used to fit distributions of parameters and summary statistics in our workflow, multivariate gaussian mixture (MGM) models, with masked autoregressive flows (MAFs), the learning method used in SLNE. MGMs gave us better results than MAFs. In our comparison, we tried two sets of control parameters of MAF training, one of which being the recommended controls in Papamakarios *et al.* (2019). The computation times of MAFs, particularly for higher-dimensional densities, prevented attempts to further tune these parameters. Hermans *et al.* (2022) also emphasized that high computation times, and the need to build a new reference table for each new dataset, restrict evaluation of coverage of iterative deep-learning based ABC methods to low-dimensional problems. However, inference of probability densities by deep-learning methods remains an active field of research. It is reasonable to anticipate further efforts directed towards the utilisation of these methods, in addition to the adjustment of their control parameters, with the objective of enhancing their performance in simulation-based inference frameworks. Our findings indicate that the older method of multivariate Gaussian mixture modelling also merits consideration.

6 Conclusion

In this study, we have implemented and evaluated summary-likelihood inference, an iterative simulation-based method for approximate likelihood inference. In addition to documenting the method's favorable performance, including in regard to confidence interval inference, we have conducted comparisons with the ABC-RF method. These comparisons demonstrate that constructing a reference table of simulations based on pre-specified priors may result in the omission of the most pertinent parameter regions. In contrast, an iterative method of exploring the parameter space may prove more effective in identifying these regions. The full inference for any given dataset required a moderate number of simulations, of the order of 3000 times the number of estimated parameters, showing that approximate likelihood infer-

ence is a practically achievable objective even when intensive simulation of genomic datasets is required.

While iterative ABC methods have the potential to outperform non-iterative simulation-based approaches such as standard rejection ABC or ABC-RF, they are less commonly used by non-experts. This may be due to factors such as simplicity, robustness, computational requirements and familiarity with non-iterative approaches. However, as the field advances and tools become more user-friendly, there should be a shift towards iterative methods, particularly for studies involving high-dimensional parameter spaces and where computational efficiency is critical.

Acknowledgments

We thank A. Courtiol for comments on the manuscript, and the Genotoul bioinformatics platform Toulouse Occitanie (`https://doi.org/10.15454/1.5572369328961167E12`) for providing computing resources. This work has been supported by funds from the Occitanie Regional Council's program "Key challenge BiodivOc" managed by the University of Montpellier (DevOCGen project), and has benefited from State aid managed by the Agence Nationale de la Recherche under the France 2030 programme, reference ANR-23-EXMA-0002 ("AgroStat" project).

Code availability

The source code of the `diyabc` simulator used in this study is available at `https://github.com/diyabc/diyabc/releases/tag/v1.1.36`. The R packages used in this study are all available on `https://cran.r-project.org/`. Additional scripts are available at `https://github.com/f-rousset/SI_InfusionMS` and their final version will be uploaded to an institutional data repository such as `https://entrepot.recherche.data.gouv.fr/dataverse/cnrs`.

References

1. J. Akeret, A. Refregier, A. Amara, S. Seehars, C. Hasner, Approximate Bayesian computation for forward modeling in cosmology. *Journal of Cosmology and Astroparticle Physics* **2015**, 043 (2015).
2. M. Appleton *et al.*, Protected area personnel and ranger numbers are insufficient to deliver global expectations. *Nature Sustainability* **5**, 1100–1110 (2022).
3. M. Auger-Méthé *et al.*, A guide to state–space modeling of ecological time series. *Ecological Monographs* **91**, e01470 (2021).
4. M. S. Bartlett, Properties of sufficiency and statistical tests. *Proc. Roy. Soc. (Lond.) A* **160**, 268–282 (1937).
5. M. A. Beaumont, Approximate Bayesian Computation in evolution and ecology. *Ann. Rev. Ecol. Evol. Syst.* **41**, 379–406 (2010).
6. M. A. Beaumont, J.-M. Cornuet, J.-M. Marin, C. P. Robert, Adaptive approximate Bayesian computation. *Biometrika* **96**, 983–990 (2009).
7. M. A. Beaumont, W. Zhang, D. J. Balding, Approximate Bayesian computation in population genetics. *Genetics* **162**, 2025–2035 (2002).
8. G. Bertorelle, A. Benazzo, S. Mona, ABC as a flexible framework to estimate demography over space and time: some cons, many pros. *Molecular Ecology* **19**, 2609–2625 (2010).
9. M. G. B. Blum, O. François, Non-linear regression models for approximate Bayesian computation. *Stat. Computing* **20**, 63–73 (2010).
10. M. G. B. Blum, M. A. Nunes, D. Prangle, S. A. Sisson, A comparative review of dimension reduction methods in approximate Bayesian computation. *Stat. Sci.* **28**, 189–208 (2013).
11. H. Bozdogan, "Mixture model cluster analysis using model selection criteria and a new informational measure of complexity" in *Proceedings of the first US/Japan conference on the frontiers of statistical modeling: an informational approach*, ed. by H. Bozdogan, pp. 69–113.
12. L. Breiman, Statistical modeling: the two cultures (with Discussion). *Statist. Sci.* **16**, 199–231 (2001).
13. G. Casella, R. L. Berger, *Statistical inference* (Duxbury, Pacific Grove, CA, 2002).

14. F.-D. Collin *et al.*, Extending approximate Bayesian computation with supervised machine learning to infer demographic history from genetic polymorphisms using DIYABC Random Forest. **21**, 2598–2613 (2021).

15. K. Cranmer, J. Brehmer, G. Louppe, The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences* **117**, 30055–30062 (2019).

16. A. C. Daly, D. Gavaghan, J. Cooper, S. Tavener, Inference-based assessment of parameter identifiability in nonlinear biological models. *Journal of The Royal Society Interface* **15**, 20180318 (2018).

17. A. C. Davison, D. V. Hinkley, *Bootstrap methods and their applications* (Cambridge Univ. Press, 1997).

18. P. Del Moral, A. Doucet, A. Jasra, An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing* **22**, 1009–1020 (2012).

19. P. Del Moral, A. Doucet, A. Jasra, Sequential Monte Carlo Samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology* **68**, 411–436 (2006).

20. P. J. Diggle, R. J. Gratton, Monte Carlo methods of inference for implicit statistical models. *J. R. Stat. Soc. B* **46**, 193–227 (1984).

21. K. N. Dinh, Z. Xiang, Z. Liu, S. Tavaré, *Approximate Bayesian Computation sequential Monte Carlo via random forests*, 2024, arXiv: `2406.15865`.

22. Y. Fan, S. R. Meikle, G. I. Angelis, A. Sitek, in *Handbook of Approximate Bayesian Computation*, ed. by S. A. Sisson, Y. Fan, M. A. Beaumont (CRC Press, 2019), pp. 623–647.

23. Y. Fan, D. J. Nott, S. A. Sisson, Approximate Bayesian computation via regression density estimation. *Stat* **2**, 34–48 (2013).

24. P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees. *Machine Learning* **36**, 3–42 (2006).

25. D. Greenberg, M. Nonnenmacher, J. Macke, "Automatic Posterior Transformation for Likelihood-Free Inference" in *Proceedings of the 36th International Conference on Machine Learning*, ed. by K. Chaudhuri, R. Salakhutdinov, vol. 97, pp. 2404–2414.

26. T. Hastie, R. Tibshirani, J. Friedman, *The elements of statistical learning: data mining, inference and prediction* (Springer, ed. 2, 2009).

27. J. Hermans, A. Delaunoy, F. Rozet, A. Wehenkel, V. Begy, G. Louppe, A crisis in simulation-based inference? Beware, your posterior approximations can be unfaithful. *Trans. Mach. Learn. Res.* (2022), arXiv: `2110.06581v3`.

28. D. P. Kingma, J. Ba, "Adam: A Method for Stochastic Optimization." in *International Conference on Learning Representations (ICLR)*, ed. by Y. Bengio, Y. LeCun.

29. R. Lebret, S. Iovleff, F. Langrognet, C. Biernacki, G. Celeux, G. Govaert, Rmixmod: The R Package of the Model-Based Unsupervised, Supervised, and Semi-Supervised Classification Mixmod Library. *Journal of Statistical Software* **67**, 1–29 (2015).

30. E. Lombaert *et al.*, Inferring the origin of populations introduced from a genetically structured native range by approximate Bayesian computation: case study of the invasive ladybird Harmonia axyridis. *Molecular Ecology* **20**, 4654–4670 (2011).

31. J.-M. Lueckmann, J. Boelts, D. S. Greenberg, P. J. Gonçalves, J. H. Macke, "Benchmarking Simulation-Based Inference" in *Proceedings of the 24th International Conference on Artifi-cial Intelligence and Statistics (AISTATS)*, ed. by A. Banerjee, K. Fukumizu, vol. 130, pp. 343–351.

32. J.-M. Lueckmann, P. J. Goncalves, G. Bassetto, K. Öcal, M. Nonnenmacher, J. H. Macke, in *Advances in Neural Information Processing Systems 30*, ed. by I. Guyon *et al.* (Curran Associates, Inc., 2017), pp. 1289–1299.

33. J.-M. Marin, L. Raynal, P. Pudlo, C. P. Robert, A. Estoup, *abcrf: Approximate Bayesian Computation via Random Forests*, R package version 1.9, URL: `https://CRAN.R-project.org/package=abcrf`.

34. G. Papamakarios, I. Murray, in *Advances in Neural Information Processing Systems 29*, ed. by D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, R. Garnett (Curran Associates, Inc., 2016), pp. 1028–1036.

35. G. Papamakarios, T. Pavlakou, I. Murray, in *Advances in Neural Information Processing Systems 30*, ed. by I. Guyon *et al.* (Curran Associates, Inc., 2017), pp. 2335–2344.

36. G. Papamakarios, D. Sterratt, I. Murray, "Sequential Neural Likelihood: Fast Likelihood-free Inference with Autoregressive Flows" in *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, ed. by K. Chaudhuri, M. Sugiyama, vol. 89, pp. 837–848.

37. B. M. Peter, D. Wegmann, L. Excoffier, Distinguishing between population bottleneck and population subdivision by a Bayesian model choice procedure. eng, *Mol. Ecol.* **19**, 4648–4660 (2010).

38. J. C. Pinheiro, D. M. Bates, Unconstrained parametrizations for variance-covariance matrices. *Statistics and Computing* **6**, 289–296 (1996).

39. D. Prangle, Adapting the ABC Distance Function. *Bayesian Analysis* **12**, 289–309 (2017).

40. P. Pudlo, J.-M. Marin, A. Estoup, J.-M. Cornuet, M. Gautier, C. P. Robert, Reliable ABC model choice via random forests. *Bioinformatics* **32**, 859–866 (2016).

41. L. Raynal, J.-M. Marin, P. Pudlo, M. Ribatet, C. P. Robert, A. Estoup, ABC random forests for Bayesian parameter inference. *Bioinformatics* **35**, 1720–1728 (2019).

42. F. Rousset, A. Gouy, C. Martinez-Almoyna, A. Courtiol, The summary-likelihood method and its implementation in the Infusion package. *Mol. Ecol. Res.* **17**, 110–119 (2017).

43. F. Rozet, F. Divo, S. Schnake, *Zuko: Normalizing flows in PyTorch*, version 1.2, URL: `https://zenodo.org/records/10571785`.

44. F. J. Rubio, A. M. Johansen, A simple approach to maximum intractable likelihood estimation. *Electron. J. Stat.* **7**, 1632–1654 (2013).

45. Y. Schälte, J. Hasenauer, Efficient exact inference for dynamical systems with noisy measurements using sequential approximate Bayesian computation. *Bioinformatics* **36**, i551–i559 (2020).

46. L. Sharrock, J. Simons, S. Liu, M. Beaumont, *Sequential Neural Score Estimation: Likelihood-Free Inference with Conditional Score Based Diffusion Models*, 2024, arXiv: `2210.04872 [stat.ML]`.

47. S. A. Sisson, Y. Fan, M. A. Beaumont, Eds., *Handbook of Approximate Bayesian Computation* (CRC Press, 2019).

48. S. Tavaré, D. J. Balding, R. C. Griffiths, P. Donnelly, Inferring coalescence times from DNA sequence data. *Genetics* **145**, 505–518 (1997).

49. The 1000 Genomes Project Consortium, An integrated map of genetic variation from 1,092 human genomes. *Nature* **491**, 56–65 (2012).

50. T. Toni, D. Welch, N. Strelkowa, A. Ipsen, M. P. Stumpf, Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of The Royal Society Interface* **6**, 187–202 (2009).

51. S. N. Wood, Statistical inference for noisy nonlinear ecological dynamic systems. *Nature* **466**, 1102–1104 (2010).

52. M. N. Wright, A. Ziegler, `ranger`: A fast implementation of random forests for high dimensional data in `C++` and `R`. *Journal of Statistical Software* **77**, 1–17 (2017).

Supplementary Material

S.1 Control parameters of algorithms

S.1.1 Iterative sampling of parameter points

The aim of the iterative procedure is to fill a target "top" space defined as the space of parameters with a given minimum likelihood ratio $l_{\min}$ relative to the summary-MLE. As a first approach, a uniform sampling of the currently predicted top space, together with sampling out of it with decreasing density for decreasing likelihood, may seem appropriate to identify iteratively the boundaries of the target space and to infer the likelihood surface within it. An approximately uniform sampling of the target space is obtained by a simple rejection sampling method, where parameter vectors are first sampled from a given proposal distribution $q_{\boldsymbol{\Theta}}$ covering the target space, and the following rejection step preferentially retains sampled points with low $q_{\boldsymbol{\Theta}}$ density, with probability in inverse proportion of the $q_{\boldsymbol{\Theta}}$ density relative to its maximum among the sampled points. The current estimate $\hat{\imath}_{\boldsymbol{\Theta}}$ of the instrumental density is used as proposal distribution, except as specified below. To explore the boundaries, distinct rejection probabilities are also computed for points outside the top region. These probabilities not only depend on the $q_{\boldsymbol{\Theta}}$ density but also on the likelihood ratios.

We have met some problems when applying such rules. From a broad initial instrumental distribution, few of the randomly sampled points have likelihood high enough to be retained after the rejection step, so fewer points may be retained than a given target number n_{i+1} unless an impractically large number of candidate points is sampled (in practice the number of candidate points generated has been bounded to essentially $10n_{i+1}$ even if the acceptance rate in previous iterations was much lower than 0.1). This problem is of particular concern in high-dimensional spaces. This can be improved by the use of the current instrumental posterior density $\hat{\imath}_{\boldsymbol{\Theta}|T}(.|\mathbf{p}[\mathbf{u}(\mathcal{D})])$ as a proposal distribution for the sampling of a fraction of the points. More precisely, the current $\hat{\imath}_{\boldsymbol{\Theta}}$ is used to obtain 90% of the points (as a target: maybe less if rejection rate is high), and the instrumental posterior is used to generate the missing points.

Another problem that was apparent, at least in the early versions of the workflow, is that these rules may sample too few points on the boundaries

of the top space. Several modifications have been implemented to favor exploration at these boundaries. In the current workflow, performance may be practically insensitive to these modifications, except in the early iterations and for cases such as those where uniform sampling of parameter space leads to poor ABC-RF performance. Firstly, points are sampled not from each of the multivariate Gaussian distributions of the MGM model, but from the corresponding multivariate Student distributions (with dispersion matrix as given by the Gaussian covariance matrix, and degree of freedom set to 1). Secondly, the proportion of retained points at the boundary can be controlled by rescaling their rejection probabilities. A rescaling of the likelihood ratio of outside points sampled from $\hat{\imath}_{\boldsymbol{\Theta}}$ has been implemented so that the proportion of outside points among retained ones can increase to a given value q. All simulations used $q = 1/4$. The rejection rule for points sampled from the instrumental posterior distribution is more complex, but also designed to sample points at the boundaries of the top space.

These rules depart from the idea of sampling from the posterior distribution, applied in deep-learning based iterative workflows. However, for uniform priors, sampling new parameters from the posterior distribution is equivalent to sampling with relative probabilities in proportion to likelihood ratios, and is then similar to a simple variant of our sampling procedure. We implemented such sampling in proportion to likelihood ratios, still with rejection sampling, and still allowing fewer points to be retained than the prespecified target number. This means that iterations still proceed by smaller steps when rejection sampling has low success rate. We compared the two versions in two test cases (Tables S.12 and S.8). With the current workflow and its default controls, the performance of the two sampling algorithms are indistinguishable. Only by focusing on parameters for which uniform sampling of parameter space leads to distinctly poor ABC-RF performance, and on the early iterations, can we detect some effect of the sampling rule(Table S.13).

S.1.2 Selection of simulated samples for projections

In each iteration i, a first subset S_i of points can be defined as follows. For each parameter point in the reference table, the predicted (summary-) log-likelihood is computed. Then all points with log-likelihood above $m - F^{-1}_{\chi^2_p}(0.95)/2$ are retained, where m is the maximum predicted log-likelihood for the table, p is the number of estimated parameters, and $F^{-1}_{\chi^2_p}$ is the quan-

tile function of the chi-squared distribution with p degrees of freedom. This criterion defines a parameter region analogous to a p-dimensional 95% confidence region, except that m is not the inferred maximum of the likelihood surface. The final subset for iteration i is the union of S_i, of S_{i-1} (as using only S_i sometimes resulted in a form of periodic behavior of the projections and likelihood inferences over successive iterations), and of the initial reference table so as to allow training over the entire parameter ranges initially considered. When these rules select less than $1000p$ points, additional points from the top are retained up to that number.

S.1.3 Control of random-forest methods

Default values are defined for control settings of random-forest methods in `Infusion` and `abcrf`, which were used in all simulations reported below. 1000 trees were used in all cases for both methods. The other default settings differ between the two methods, as those from `Infusion` combine features of Breiman's original random-forest method and of the extremely-randomized trees method (Geurts *et al.*, 2006). The latter method does not provide out-of-bag predictions, but the combined options used as defaults by `Infusion` do provide such predictions. This combination is made possible by the `ranger` R package (Wright *et al.*, 2017), and the combined defaults (`splitrule="extratrees"`, `mtry` set to the number of predictor variables, `replace` set to `FALSE`, and `sample.fraction` set to 0.632) were chosen in light of the results of an independent simulation study of the effects of the settings on prediction performance (Appleton *et al.*, 2022, Appendix), giving some slight advantage to these combined settings over the basic random-forest method.

S.1.4 Summary-likelihood maximization

Practical issues in the maximization of the inferred likelihood surface are the possible occurrence of local maxima, and premature termination of maximization procedures at a boundary of the parameter space. Such occurrences may become manifest when computing likelihood profiles and summary-LRTs, where it may appear that the inferred likelihood maximum was in fact only a local maximum, exceeded by the profile likelihood for some other parameter values. Although the issue of local maxima does not have a perfect solution, it is possible to draw many candidate vectors $\boldsymbol{\theta}$ from the "instrumental

posterior" distribution $\hat{\imath}_{\boldsymbol{\Theta}|T}(.|\mathbf{p}[\mathbf{u}(\mathcal{D})])$, derived from the current joint density estimate $\hat{\mathrm{P}}_{T,\boldsymbol{\Theta}}$ by conditioning on $T = \mathbf{p}[\mathbf{u}(\mathcal{D})]$. This posterior distribution is easily deduced from the joint MGM model for the parameters and the projected summaries. Using the candidate vector $\boldsymbol{\theta}$ with the highest predicted likelihood as the initial value for maximization appears to be relatively efficient in reducing the impact of local maxima on the final inference.

An additional problem of maximization steps is that a spurious maximum of the likelihood function may be inferred by extrapolation in poorly sampled parameter regions. To avoid this, a penalization has to be defined for likelihood predictions in such regions. This penalization compares the predicted parameter density of the parameter point whose likelihood is to be evaluated, to a threshold defined from the distribution of fitted parameter densities for sampled points at the top of the likelihood surface.

S.1.5 Control of multivariate Gaussian mixture modeling

The number of components K of the MGM models must be selected. Akaike's information criterion (AIC) may be used to select the number of components, which seems appropriate given that the retained model is used for prediction of modelled density. Other model selection criteria are often considered in the MGM literature, but for different purposes, such as estimating a "true" number of components of the mixture. Lebret *et al.* (2015) suggest trying K up to $n^{0.3}$ components (see Bozdogan, 1994, p. 77, for discussion), where n is the number of points (here, the number of sample simulations in the reference table). For small-dimensional distributions, preliminary simulations showed that higher values, accessible by a minor increase of the exponent, may improve the `Infusion` inference for large n. However, when applying such rules for given n and increasing dimension of the inferred distribution, the number P of parameters of the MGM model will exceed n, highlighting that a lower bound must be considered. So we defined two constraint on K: it is bounded by the constraint $P \leq n/4$, and by $\lceil n^{0.31} \rceil$. Given such rules for the maximum number of components, we found that for large reference tables (in terms of number of simulations times number of variables), the largest value is generally retained according to the AIC criterion. Thus, it is practically equivalent, but much less time-consuming than trying all possible numbers of components, to systematically set K to the upper bound defined by the above two constraints. This simple rule was applied systematically beyond the first iteration.

Given these controls, the other controls of the MGM fitting algorithms were set to their default values in the `Rmixmod` package, except that the maximum number of iterations of the Expectation-Maximization algorithm was increased to 1000, and that non-default initialization values may be used. The MGM fit may occasionally fail or return inappropriate results, such as a mixture containing a Gaussian component with vanishing variances. In such cases, the model is refitted using one of the alternative initialization methods implemented in `Rmixmod`, and if this still fails, the number of components of the mixture is reduced.

S.1.6 Control of iterative refinements

A target number n_I of samples controls the number of samples n_i added in any iteration i, i.e., between each MGM fit of the reference table. n_i may sometimes be lower than n_I depending on realized success frequency in rejection sampling. n_{I} varied between 500 and 625 for the different simulations in this work. The main drawback of increasing the number of iterations by reducing n_{I} for a given final number of points is the increasing computation time due to the MGM modelling performed in each iteration (with a significant increase for large dimensional distributions only) and to projection steps (in many if not all iterations, see Section Iterative refinements in the Main Text). Typical computation times are presented in Table S.1.

The number of points added in the first iterations are controlled in a distinct way from later ones because it is better to sample comparatively fewer points per iteration when the likelihood surface is yet relatively poorly estimated. We used at most $n_1 = 1000$ in the first iteration, then added samples over several iterations by steps of n_i samples increasing essentially in proportion to powers of two over these iterations.

For example, for 7 estimated parameters, the first reference table contains 400 samples, then successive target numbers of added samples in the next 5 iterations are 62, 62, 125, 250, 501 to reach a total of 1400 samples. The target n_i in further iterations is 500, and iterations are performed until the reference table contains 20,000 samples. In practice, users do not have to set such values, which are default values set by the software as function of the number of estimated parameters.

scenario	parameters	total	`diyabc`	`ranger`	`Rmixmod`
Human admixture	7	2118	610	900	576
Ladybird invasion	8	2815	60	1217	1504
Human admixture	13	13691	1241	7555	4728
toy Normal	15	5949	NA	NA	5838

Table S.1: Approximate fit times (in seconds)

These are the elapsed ("wall-clock") times reported by the R code profiler, for fitting a single dataset using the default workflow design, with details for the three main components: sample simulation by `diyabc` simulator, random-forest analyses by `ranger` R package, and MGM modelling by `Rmixmod` R package. The first two steps used the parallelization procedures implemented in C++ in each software, with 10 CPU cores. The analyzed datasets in these fits were either the real data for the scenarios of historical demography (the variant with composite bottleneck parameters for the 13-parameter case), or the first simulated dataset of the toy multivariate normal example. All computations were run on a virtual Linux machine, on a Dell Precision 3581 laptop.

S.1.7 Controls of deep learning using masked autoregressive flows

This method is controlled by several parameters. Results for two sets of controls with similar computation times, "PSM19" and "alternative", are presented. In the first one, control values were set as recommended by Papamakarios *et al.* (2019, Section 5.1): they include the number of "autoregressive layers" $K = 5$, the number of "hidden layers" $L = 2$, the number of units $H = 50$ per hidden layer, and the following control parameters of the Adam optimizer (Kingma *et al.*, 2015): minibatch size = 100, learning rate = 10^{-4}, validation patience = 20, and fraction of sample simulations used as validation set= 5%. The following controls differ in the alternative set: $K = 3$, learning rate = 10^{-3}, validation patience = 30, and H. The latter is set so that the number of parameters of the MAF model (as approximated in Papamakarios *et al.*, 2017, Supplementary Material Table 1) is a power of two, and is within 8 to 16 times the number of parameters of the MGM models applied by default to reference tables of the same size. In the following simulations, this rule gives $H = 128$.

While it is possible to run the full inference workflow using MAFs rather

than MGM models, we can assess the effect of using MAFs to learn the likelihood surface in a less computer-intensive way, by training conditional MAFs to learn these surfaces only on the final reference tables of the MGM-based inferences. We performed this computation for the first 200 simulated datasets and the associated reference tables obtained when fitting the 7-parameter model to each of these datatets. This MAF training (with "PSM19" controls) takes 449 s on average for each of the first 20 reanalyzed reference tables when performed in the same computation environment as in Table S.1, using `pytorch-cuda` on an NVIDIA RTX A500 Laptop GPU. It therefore approaches the total time spent in fitting MGM models in all iterations, for the same 7-parameter inferences (576 s, Table S.1). In a full workflow using MAFs, the MAFs would be trained in each iteration. In addition, other densities needed, e.g. to sample new parameter points, would have to be estimated using either MAFs or MGM models.

S.1.8 Simplified bootstrap procedure

We used the parametric bootstrap to simulate the distribution of the summary-likelihood ratio statistic and to deduce from it a corrected p-value. We used it also to compute bootstrap confidence intervals. In its canonical version, the parametric bootstrap would be too time-consuming, as it would imply repeating the full inference workflow on many simulated datasets. However, in the present case, we used two approximations. First, it is possible to obtain the parametric bootstrap samples directly as a vector of projected summary statistics, denoted $\mathbf{t}^*$, since the distribution of such samples, given the parameter estimates as in standard parametric bootstrap, is a MGM distribution easily deduced from the representation of the joint distribution stored in the fit object. Second, we did not reconstruct references tables for each of these bootstrap samples. Instead, inferences are performed on $\mathbf{t}^*$, using the available estimated joint and marginal distributions to evaluate the terms in the expression for the summary-likelihood of the bootstrap sample, $L(\boldsymbol{\theta}; \mathbf{t}^*) = \mathrm{P}_{T,\boldsymbol{\Theta}}(\mathbf{t}^*, \boldsymbol{\theta})/\mathrm{P}_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$.

Only 199 bootstrap samples were used for each bootstrap computations. It is advised to use more samples when analyzing a given dataset. However, such a low number is enough to appreciate the correction brought by the bootstrap procedure, and there is no point in obtaining very accurate tests and intervals from many bootstrap samples for each simulated dataset, when coverage is evaluated from 200 or 400 datasets.

S.2 15-parameter toy example

S.2.1 Details of simulation scenarios

In this example, the parameter space is the space of 5×5 covariance matrices. For easy exploration of this parameter space, a transformation was applied to the parametric matrices (but not to the simulated statistics), so that the transformed space is unconstrained (Pinheiro *et al.*, 1996): transformed parameter values are the elements of the upper-triangular Cholesky factor $\mathbf{U}$ of the covariance matrix $\mathbf{V}$ ($\mathbf{V} = \mathbf{U}^\top \mathbf{U}$).

We defined a variant of this model where only 5 of the 15 parameters are identifiable, by performing simulations in which the covariance matrix is the cross-product of the matrix retaining only the diagonal elements of the 15-parameter Cholesky factor. In such a way, only diagonal covariance matrices are produced, and only the 5 standard deviation parameters (on diagonal positions of the Cholesky factor) are identifiable.

For each of these two variants, 400 simulated data were generated for the following transformed parameters, presented as the elements of the upper triangle of the Cholesky factor:

$$\mathbf{U} = (u_{ij}) = \begin{pmatrix} 2.603 & -1.689 & -0.1089 & 1.280 & 2.528 \\ & 3.670 & -0.1172 & -1.727 & 0.5490 \\ & & 3.522 & 1.690 & 0.2382 \\ & & & 2.578 & -1.048 \\ & & & & 2.592 \end{pmatrix}. \tag{S.1}$$

The corresponding data-generating covariance matrix $\mathbf{V}^\dagger$ was obtained as a draw from a Wishart distribution with unit covariance matrix and 10 degrees of freedom. Parameters of the reference table were sampled in the ranges $[0.5, 6]$ for the diagonal elements of the Cholesky factor, and $[-4, 4]$ for the non-diagonal elements. These ranges were chosen to contain all values of Cholesky factors of 1000 samples from the same Wishart distribution as used to generate the above matrix.

S.2.2 Additional simulation results

parameter	rel. bias	rel. RMSE	profLR	bootLR	BcorCI
u_{11}	0.00328	0.048	0.97	0.97	**0.968**
u_{12}	-0.00392	0.0713	**0.948**	0.962	0.96
u_{22}	-0.00183	0.0638	**0.968**	0.98	0.97
u_{13}	0.00682	0.0634	**0.962**	0.975	0.965
u_{23}	-0.00121	0.0623	**0.962**	0.982	0.97
u_{33}	-0.0083	0.0692	0.92	**0.942**	0.94
u_{14}	0.00275	0.0634	**0.948**	0.972	0.97
u_{24}	0.00097	0.0583	**0.952**	0.96	0.955
u_{34}	-0.00381	0.0501	0.955	0.965	**0.952**
u_{44}	-0.0142	0.0503	0.938	**0.952**	0.945
u_{15}	0.00606	0.0605	**0.972**	0.978	0.972
u_{25}	0.000924	0.0524	**0.952**	0.968	0.96
u_{35}	-0.00181	0.0501	**0.952**	0.965	0.96
u_{45}	0.00641	0.0463	**0.945**	0.96	0.942
u_{55}	-0.0186	0.0505	0.938	**0.945**	0.945
u_{11}	-0.00132	0.0512	**0.948**	0.948	0.948
u_{22}	0.00427	0.0648	**0.972**	0.982	0.982
u_{33}	0.00359	0.0692	**0.952**	0.955	0.952
u_{44}	-0.000685	0.0499	**0.958**	0.962	0.96
u_{55}	0.000772	0.0456	**0.965**	0.972	0.972

Table S.2: Performance of inferences for multivariate Gaussian toy model . Last five lines are for the model where only the variances are identifiable.

S.3 Ladybird invasion scenario

S.3.1 Details of simulation scenarios

This scenario, described in Fig. 1 (left panel) of the Main Text, is derived from the study by Lombaert *et al.* (2011) on the worldwide invasion routes of the harlequin ladybird *Harmonia axyridis*, based on genotypes obtained at 18 microsatellite loci. More specifically, the scenario describes the genetic origin of the invasive West-European population ("Pop 4"; sample size: 32 diploid individuals) observed for the first time in 2001 and resulting from a genetic mixture between individuals originating from the invasive population of the

Eastern USA ("Pop3": 34 sampled individuals) and individuals originating from a European bio-control population ("Pop2": 29 sampled individuals), the last two populations having been introduced from the species' native area (i.e. China; "Pop1": 29 sampled individuals) at known times (60 and 48 generations in the past for Pop2 and Pop3, respectively). A similar scenario was already taken as an example for ABC-RF analyses by Pudlo *et al.* (2016). The version studied here differs slightly from that of Lombaert *et al.* (2011), as the bottleneck events following each introduction are not modelled and are included in the effective sizes associated with each introduced population. On the other hand, the date of admixture t_1 of the West-European population (Pop 4) is considered as a variable parameter of the model. In the reference tables used for our inferences, the raw microsatellite genotype data for the scenario studied are summarized by a total of 76 statistical summaries described in section 2.6.1 of the `diyabc` documentation (Collin *et al.*, 2021; `https://diyabc.github.io/doc/`).

The assumed mutation model for the microsatellite markers is a generalized stepwise mutation model with two parameters, as implemented in `diyabc`: a geometric mean scale parameter $\bar{p}$ and a mean mutation probability parameter $\bar{\mu}$. Both of these parameters control the distribution of locus-specific parameters: for each locus in each sample simulation, the locus-specific mutation parameter is drawn in a truncated gamma distribution with mean $\bar{\mu}$ and shape 2, restricted to the range $(10^{-6}, 10^{-2})$; and the distribution of the number of repeats X of the microsatellite motif removed or added by a mutation event follows a geometric distribution $P(X = k) = (1 - p)p^{k-1}$ where each locus-specific p is drawn from a truncated gamma distribution with mean $\bar{p}$ and shape 2, restricted to the range $(0, 0.9)$.

The ranges of sampled parameter space were set to $[-2, 1]$ for all $\log(N_{\mathrm{i}}\bar{\mu})$, $[3, 45]$ for t_1, $[0.05, 0.95]$ for r_{a}, $[-5, -2]$ for $\log(\bar{\mu})$, and $[0.01, 0.05]$ for $\bar{p}$.

The values of the parameters used to generate simulated data are shown in Table S.3 for the different studied cases.

$\log(N_1\bar{\mu})$	$\log(N_2\bar{\mu})$	$\log(N_3\bar{\mu})$	$\log(N_4\bar{\mu})^*$	t_1	r_{a}	$\log(\bar{\mu})$	$\bar{p}$	used in
0	0	0	0	13	0.5	-3	0.25	Table 1, Fig. S.1
-0.031	-0.84	0.56	-0.88	19	0.5	-2.7	0.22	Tables S.5, S.8
-0.031	-1.5	0.75	0	9	0.2	-2.7	0.22	Tables S.6, S.7

Table S.3: The three different parameter vectors $\boldsymbol{\theta}^{\dagger}$ used to generate simulated datasets in the 8-parameter examples of the ladybird invasion scenario. *: $\log(N_4\bar{\mu})$ was variable among simulated datasets in Main Text Fig. S.2, and in supplementary Tables S.4, S.7 and Fig. S.3.

S.3.2 Additional simulation results for the ladybird invasion scenario:

S.3.2.1 Distribution of p-values for first set of parameter values

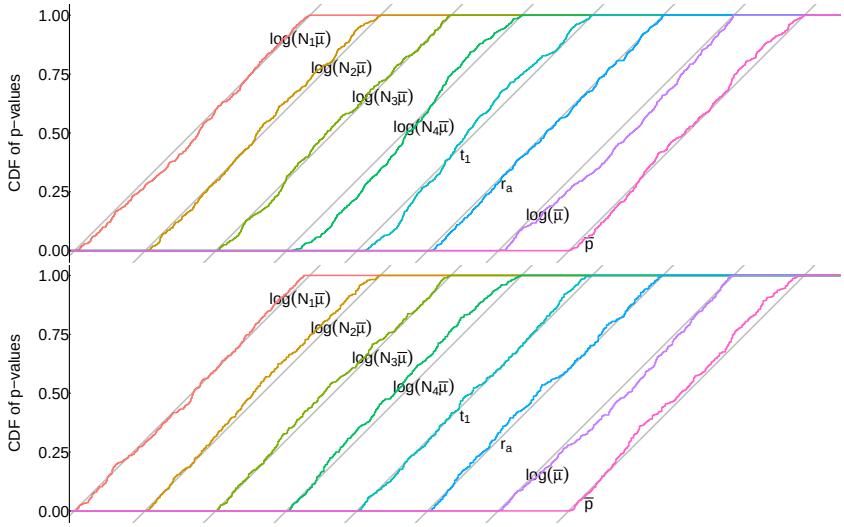

Figure S.1: Cumulative distributions of p-values for the ladybird invasion scenario

This Figure describes the same simulation results as Main Text Table 1. The distributions of each uncorrected (top) and bootstrap-corrected (bottom) summary-LRTs ("bootLR" in Tables) for each parameter is shown relative to its own 1:1 reference line.

S.3.2.2 Performance summaries for variable $\log(N_4\bar{\mu})$

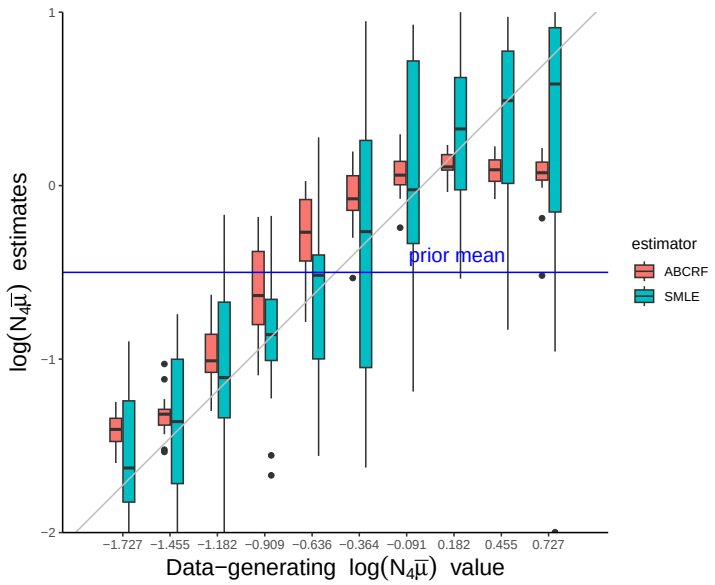

Figure S.2: Distributions of SMLE and ABC-RF estimates of the parameter $\log(N_4\bar{\mu})$. Twenty simulated samples were analyzed for each given $\log(N_4\bar{\mu})$ value. The first set of parameter values is used for the other parameters, as in Table 1.

	Relative bias			Relative RMSE			coverage				
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI	HPD CI
$\log(N_1\bar{\mu})$	**0.000456**	0.00972	0.0087	**0.0354**	0.037	0.0369	0.975	**0.945**	0.945	0.96	0.995
$\log(N_2\bar{\mu})$	**0.00482**	-0.0391	-0.0284	**0.0904**	0.12	0.123	0.98	0.965	**0.955**	0.99	0.985
$\log(N_3\bar{\mu})$	**0.00401**	-0.0284	-0.016	**0.0939**	0.119	0.126	0.98	0.965	**0.95**	0.995	0.995
$\log(N_4\bar{\mu})$	**0.00713**	0.021	0.0375	0.192	**0.121**	0.122	0.985	**0.945**	0.965	1	1
t_1	**0.0925**	0.208	0.182	0.318	0.241	**0.236**	0.995	**0.96**	0.965	0.995	0.98
r_{a}	0.0129	0.0144	**0.0102**	0.248	**0.107**	0.136	0.975	0.955	**0.95**	1	1
$\log(\bar{\mu})$	**-0.0148**	-0.0789	-0.0594	**0.0634**	0.111	0.0963	0.98	0.96	**0.955**	0.99	0.985
$\bar{p}$	**0.0195**	0.0305	0.0235	0.2	**0.157**	0.16	0.935	0.935	0.925	**0.96**	0.93

Table S.4: Performance summaries for first set of parameter values and variable data-generating $\log(N_4\bar{\mu})$. $\log(N_4\bar{\mu})$ parameter values were varied as shown in Main Text Fig. S.2. $\log(N_4\bar{\mu})$ performance summaries are here average values over these variable $\log(N_4\bar{\mu})$ values. Other parameter values were constant as given by the first set of parameter values in Table S.3. HPD CI: Highest posterior density credible intervals. The meaning of other columns is an the Main Text tables (see Table 1).

In the following Tables we first present simulation results for two sets of data-generating parameter values differing from those of Table 1 in the Main Text, in order to check whether patterns in need of specific explanations persist in these different simulation conditions.

S.3.2.3 Second set of parameter values

Here, the data-generating values (Table S.3, second line) were chosen from a preliminary fit of the actual data. Nevertheless, the results are overall similar to those of the Main Text, with even less information about parameters t_1 and r_{a}.

	Relative bias			Relative RMSE			coverage				
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI	HPD CI
$\log(N_1\bar{\mu})$	**0.00689**	0.0128	0.0115	**0.0369**	0.0376	0.0373	0.965	0.96	**0.95**	0.94	0.99
$\log(N_2\bar{\mu})$	**-0.00747**	-0.0691	-0.0634	**0.0499**	0.0857	0.0865	0.965	**0.945**	0.93	1	1
$\log(N_3\bar{\mu})$	**0.0195**	-0.0508	-0.0301	0.0996	0.0879	**0.0866**	0.98	0.94	**0.95**	1	1
$\log(N_4\bar{\mu})$	**-0.00848**	-0.0368	-0.0455	0.132	**0.0711**	0.0782	0.985	**0.95**	0.96	1	1
t_1	**0.0651**	0.151	0.157	0.285	**0.16**	0.173	1	0.96	**0.95**	1	1
r_{a}	**0.0839**	0.103	0.0999	0.243	**0.147**	0.157	0.995	**0.975**	0.98	1	0.985
$\log(\bar{\mu})$	**-0.00764**	-0.0662	-0.0552	**0.052**	0.0871	0.082	0.975	**0.95**	0.945	1	1
$\bar{p}$	**0.00925**	0.0266	0.0137	0.152	**0.142**	0.144	0.965	0.94	**0.945**	0.98	0.965

Table S.5: Performance summaries for invasive ladybird scenario for second set of data-generating values.

$\log(N_2\bar{\mu})$ ABC-RF estimates are biased unexpectedly away from the prior mean. $\log(N_3\bar{\mu})$, and to some extent r_{a} ABC-RF estimates, behave as those of low-information parameters, with a bias toward the prior. $\log(N_4\bar{\mu})$ ABC-RF estimates have low bias, remarkably low variance and are uncorrelated to the SL estimates, as in the case from the Main Text.

S.3.2.4 Third set of parameter values

This case is a variant of the previous one where the data-generating values for $\log(N_2\bar{\mu})$, $\log(N_3\bar{\mu})$, $\log(N_4\bar{\mu})$, t_1 and r_{a} were modified (Table S.3, third line).

	Relative bias			Relative RMSE			coverage				
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI	HPD CI
$\log(N_1\bar{\mu})$	**-0.00234**	0.00685	0.00555	**0.0352**	0.0359	0.0354	0.975	0.955	**0.95**	0.98	0.995
$\log(N_2\bar{\mu})$	**-0.0159**	-0.028	-0.0374	0.0506	**0.0426**	0.0494	0.99	0.975	**0.965**	1	1
$\log(N_3\bar{\mu})$	**-0.00964**	-0.124	-0.106	**0.089**	0.136	0.12	0.985	0.965	**0.955**	1	1
$\log(N_4\bar{\mu})$	**0.0404**	0.0573	0.0759	0.263	**0.0615**	0.0787	0.995	**0.97**	0.99	1	1
t_1	**0.0357**	0.248	0.201	**0.187**	0.256	0.224	0.995	**0.975**	0.98	1	1
r_{a}	**0.0111**	0.058	0.0279	**0.0448**	0.0755	0.0536	0.965	**0.94**	0.94	1	1
$\log(\bar{\mu})$	**-0.0142**	-0.0431	-0.0389	**0.0513**	0.0628	0.0602	0.96	**0.945**	0.94	1	1
$\bar{p}$	**0.0136**	0.0281	0.0191	0.153	**0.141**	0.144	0.975	0.955	**0.95**	0.985	0.975

Table S.6:
Performance summaries for the invasive ladybird scenario for third set of parameter values.

The Results are similar to those with the original data-generating values. In particular, the observations about $\log(N_4\bar{\mu})$ estimation are repeated. So we performed simulations for variable $\log(N_4\bar{\mu})$ as in the Main Text, using the third set of values of other data-generating parameters. General patterns reported in the next Fig. S.3 and Table S.7 and are similar to those for the first set of simulation parameters (Table S.4, Main Text Fig. S.2).

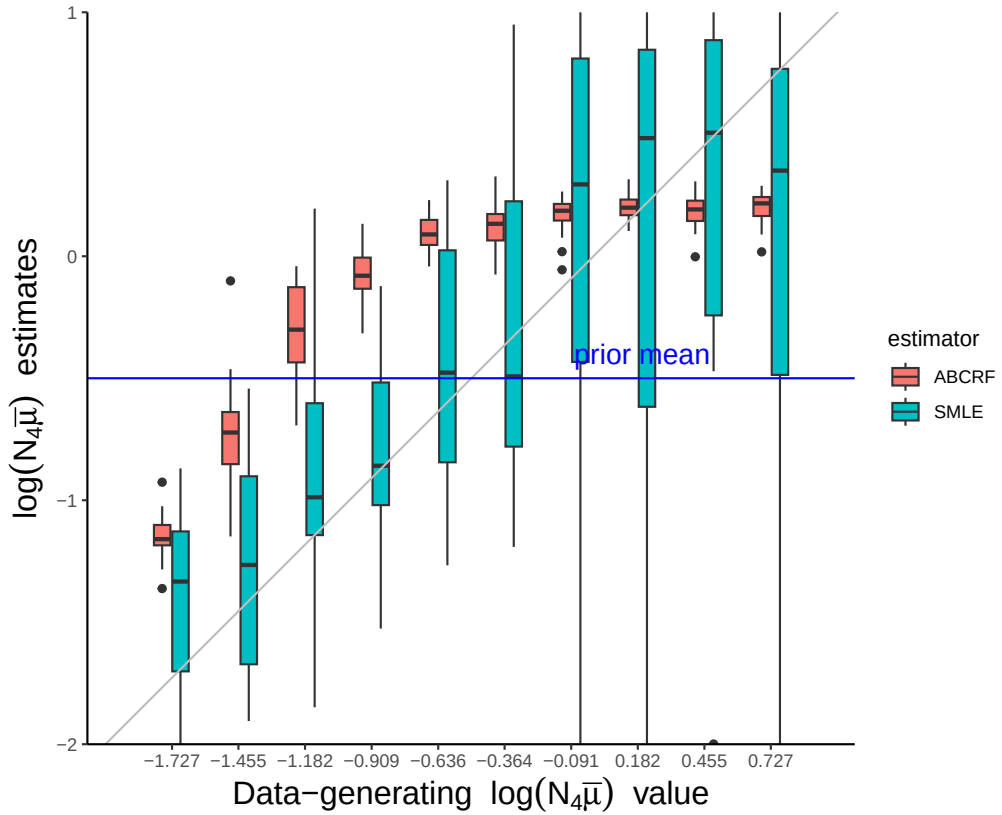

Figure S.3: Distributions of SMLE and ABC-RF $\log(N_4\bar{\mu})$ estimates for the third set of parameter values. Twenty simulated samples were analyzed for each given $\log(N_4\bar{\mu})$ value. Other parameter values were constant as given by the third set of parameter values in Table S.3.

	Relative bias			Relative RMSE			coverage				
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI	HPD CI
$\log(N_1\bar{\mu})$	**-0.0022**	0.00549	0.00426	0.0376	0.0369	**0.0364**	**0.945**	0.93	0.93	0.98	0.99
$\log(N_2\bar{\mu})$	**-0.013**	-0.0235	-0.0313	0.0557	**0.0449**	0.0514	**0.935**	0.935	0.93	0.985	0.99
$\log(N_3\bar{\mu})$	**-0.000523**	-0.116	-0.0978	**0.087**	0.129	0.112	0.975	**0.96**	0.96	1	1
$\log(N_4\bar{\mu})$	**0.0131**	0.123	0.136	0.241	**0.201**	0.208	1	**0.97**	0.99	1	0.99
t_1	**0.103**	0.326	0.306	**0.266**	0.347	0.346	0.995	0.965	0.975	**0.955**	0.935
r_{a}	**0.0206**	0.091	0.0616	**0.0953**	0.132	0.116	0.96	0.94	**0.945**	0.985	0.985
$\log(\bar{\mu})$	**-0.0106**	-0.0366	-0.0319	**0.05**	0.06	0.0581	0.96	0.94	**0.945**	1	1
$\bar{p}$	**0.0373**	0.0482	0.0403	0.171	**0.158**	0.163	0.96	0.945	0.935	0.965	**0.95**

Table S.7: Performance summaries for the invasive ladybird scenario for the third set of parameter values and variable $\log(N_4\bar{\mu})$.
$\log(N_4\bar{\mu})$ parameter values were varied as shown in Main Text Fig. S.3. $\log(N_4\bar{\mu})$ performance summaries are here average values over these variable $\log(N_4\bar{\mu})$ values. Other parameter values were constant as given by the third set of parameter values in Table S.3.

S.3.2.5 Alternative parameter sampling rule

	Relative bias			Relative RMSE			coverage				
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI	HPD CI
$\log(N_1\bar{\mu})$	**0.00735**	0.0128	0.0115	0.0374	0.0376	**0.0373**	0.975	**0.955**	0.955	0.94	0.99
$\log(N_2\bar{\mu})$	**-0.00622**	-0.0691	-0.0634	**0.0505**	0.0857	0.0865	**0.955**	0.925	0.925	1	1
$\log(N_3\bar{\mu})$	**0.0204**	-0.0508	-0.0301	0.0996	0.0879	**0.0866**	0.975	**0.955**	0.955	1	1
$\log(N_4\bar{\mu})$	**-0.0114**	-0.0368	-0.0455	0.14	**0.0711**	0.0782	0.98	**0.955**	0.96	1	1
t_1	**0.0668**	0.151	0.157	0.304	**0.16**	0.173	**0.96**	0.92	0.94	1	1
r_{a}	**0.0946**	0.103	0.0999	0.25	**0.147**	0.157	0.985	**0.975**	0.975	1	0.985
$\log(\bar{\mu})$	**-0.00623**	-0.0662	-0.0552	**0.0516**	0.0871	0.082	0.97	**0.945**	0.935	1	1
$\bar{p}$	**0.0119**	0.0266	0.0137	0.157	**0.142**	0.144	0.975	0.97	**0.965**	0.98	0.965

Table S.8: Performance summaries with alternative parameter sampling rule, for the invasive ladybird scenario.
These results are based on running the inference workflow on the same 200 datasets as in Table S.5, simulated under the second set of parameters.

S.4 Human admixture scenario

S.4.1 Details of simulation scenario

S.4.1.1 The demo-genetic model

The Human admixture scenarios considered in our study are sub-models of a scenario with 17 variable parameters studied by Raynal *et al.* (2019) and Collin *et al.* (2021, scenario 2) to illustrate the ABC-RF method. This scenario includes (i) an ancient demographic change in the ancestral African population, (ii) a single out-of-Africa colonization event giving an ancestral out-of-Africa population which secondarily split into one European and one East Asian population, and (iii) a recent genetic admixture of the Americans of African Ancestry in SW-USA between their African ancestors and individuals of European origin. The original 17-parameter scenario is detailed in Figure 1 (right panel) of the Main Text. In this highly parameterized scenario, very few original parameters can be estimated. For this reason, we have chosen to derive sub-models in which we have fixed the number of variable parameters to 7 or 13, by fixing 10 or 4 parameters respectively, as detailed below.

Other notable differences compared to Raynal *et al.* (2019) lie in the way by which the event times have been expressed and in the way the `diyabc` software is used. In its basic use, the `diyabc` software draws values from parameter prior distributions until they satisfy a given set of constraints on the parameters (such as $t_1 < t_2 < t_3 < t_4$ in our scenario). A non-standard use of the `diyabc` simulator (based on the `-o` option in the online command of the executable file) is to provide it with a table of parameter values that do not depend on its rejection sampling and are therefore fully controlled by the user, possibly with different rules than the rejection sampling described above. Further, one can re-parametrize the model according to differences between split or merge time events $t_{12} = t_2 - t_1$, $t_{23} = t_3 - t_2$, $t_{34} = t_4 - t_3$, and sample these new parameters only under the constraint that they are positive. This allows more transparent control of the distribution of sampled parameter points by the inference workflow. Hence this reparametrization was used for SL inferences.

The choice of either sampling rules (rejection versus re-parametrization) has different implications for the two inference methods: rejection sampling under the constraint that $t_1 < t_2 < t_3 < t_4$ means that the sampled prior

distributions of parameters are not uniform and have relatively low density near the boundaries defined by the non-box constraints (e.g., near $t_2 = t_3$). This tends to push posterior-based point estimates away from the boundary (i.e. when using ABC-RF), in comparison to posterior estimates based on uniform priors. Using the reparametrization instead of rejection sampling avoids this effect. The maximum-likelihood estimates obtained by our summary-likelihood method are comparatively insensitive to this effect of rejection sampling and can be much closer to the boundary. Hence, for easier comparison of SL and ABC-RF results, the reparametrization was also used for building the reference tables used in ABC-RF inferences.

The SNP data associated to those human admixture scenarios were obtained from individuals originating from four Human populations (30 unrelated individuals per population) using the freely accessible public 1000 Genome databases (i.e. the VCF format files including variant calls available at `http://www.1000genomes.org/data`; The 1000 Genomes Project Consortium, 2012). The four Human populations included the Yoruba population (Nigeria), encoded YRI in the 1000 genome database; the Han Chinese population (China), encoded CHB; the British population (England and Scotland), encoded GBR; and the population composed of Americans of African Ancestry in SW-USA, encoded ASW. Five or ten thousands independent SNP loci were selected from the 22 autosomal chromosomes using the criteria described in the Appendix S2 of Collin *et al.* (2021). In particular, only SNPs with a minimum allele frequency ≥ 0.01 were retained, and the same constraint was applied in the `diyabc` simulations. In the reference tables used for our inferences, the raw SNP genotype data for the scenarios studied are summarized by a total of 130 statistical summaries described in section 2.6.3 of the `diyabc` documentation (Collin *et al.*, 2021; `https://diyabc.github.io/doc/`).

S.4.1.2 Data-generating parameter values

For the 13-parameter variant of the Human admixture scenario with composite bottleneck parameters, the data-generating values for the estimated parameters are $\log(N_1) = 4.7$, $\log(N_2) = 3.4$, $t_1 = 6$, $r_{\mathrm{a}} = 0.21$, $\log(t_{12}) = 2.2$, $d_3/\mathrm{Nbn}_3 = 0.26$, $\log(\mathrm{Nbn}_3) = 2.1$, $d_4/\mathrm{Nbn}_4 = 0.1$, $\log(\mathrm{Nbn}_4) = 2.3$, $\log(1+t_{23}) = 1.3$, $d_{34}/\mathrm{Nbn}_{34} = 0.44$, $\log(\mathrm{Nbn}_{34}) = 1.5$, $\log(N_{\mathrm{a}}) = 2.7$, and those for known, non-estimated parameters are $N_3 = 6310$, $N_4 = 3162$ $N_{34} = 1259$ and $t_4 = 1007$.

For the other 13-parameter variant, without composite parameters, the data-generating values are $\log(N_1) = 3.2$, $\log(N_2) = 3.5$, $\log(N_3) = 3.8$, $\log(N_4) = 3.5$, $t_1 = 6$, $r_\mathrm{a} = 0.17$, $\log(t_{12}) = 2.2$, $d_3 = 42$, $d_4 = 9$, $\log(1+t_{23}) = 1.7$, $d_{34} = 24$, $\log(1 + t_{34}) = 2.9$, $\log(N_\mathrm{a}) = 2.7$ for estimated ones; and $N_\mathrm{bn3} = 160$, $N_\mathrm{bn4} = 98$, $N_{34} = 1259$, and $N_\mathrm{bn34} = 74$ for fixed known ones.

For the 7-parameter variant of the Human admixture scenario, the data-generating values are $\log(N_2) = 3.5$, $t_1 = 6$, $\log(t_{12}) = 2.2$, $\log(1+t_{23}) = 1.5$, $\mathrm{Nbn}_{34} = 65$, $\log(1 + t_{34}) = 3$, $\log(N_\mathrm{a}) = 2.7$ for estimated ones, and $N_1 = 12589$, $N_3 = 19953$, $N_4 = 3162$, $r_\mathrm{a} = 0.2$, $d_3 = 42$, $\mathrm{Nbn}_3 = 160$, $d_4 = 9$, $\mathrm{Nbn}_4 = 98$, $N_{34} = 1259$, $d_{34} = 24$ for fixed known ones.

S.4.1.3 Sampled parameter ranges for Human admixture scenarios

In the 7-parameters scenario, the ranges of sampled parameter space were set to $[3, 5]$ for $\log(N_2)$, $[1, 30]$ for t_1, $[\log(50), \log(5000)]$ for $\log(t_{12})$, $[0, \log(5001)]$ for $\log(1 + t_{23})$ and $\log(1 + t_{34})$, $[5, 500]$ for N_bn34, and $[2, 4]$ for $\log(N_\mathrm{a})$.

In the 13-parameters scenario without composite bottleneck parameters, additional ranges were set to $[3, 5]$ for all $\log(N_i)$, $[1, 50]$ for d_3 and d_4, and $[0, 50]$ for d_{34}.

In the 13-parameters scenario with composite bottleneck parameters, the ranges were $[3, 5]$ for $\log(N_1)$ and $\log(N_2)$, $[1, 30]$ for t_1, $[0.05, 0.95]$ for r_a, $[\log(50), \log(900)]$ for $\log(t_{12})$, $[0, 1]$ for the three composite bottleneck parameters, $[\log(50), \log(5000)]$ for $\log(N_\mathrm{bn3})$, $\log(N_\mathrm{bn4})$, and $\log(N_\mathrm{bn34})$, $[0, \log(901)]$ for $\log(1 + t_{23})$, and $[2, 4]$ for $\log(N_\mathrm{a})$.

S.4.2 7-parameter inferences: additional simulation results

S.4.2.1 Distributions of p-values

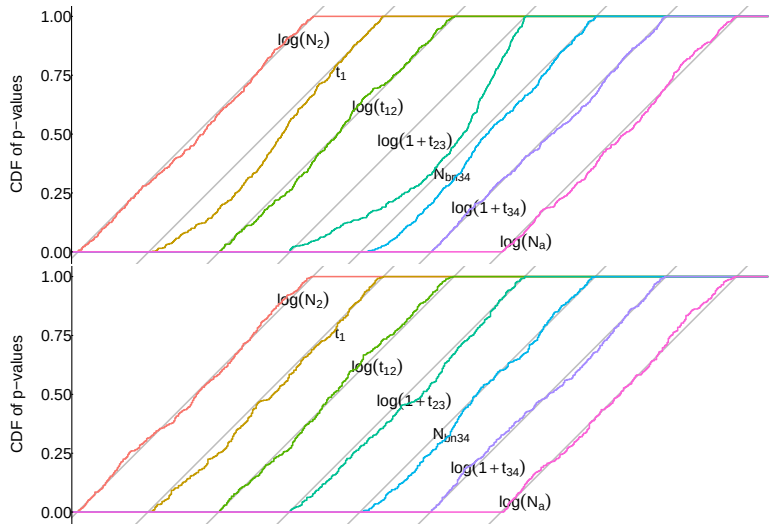

Figure S.4: Cumulative distributions of p-values of summary-LRTs for the 7-parameter human admixture scenario.
This Figure describes the same simulations as in Main Text Table 2. The distributions of each uncorrected (top) and bootstrap-corrected (bottom) tests for each parameter is shown relative to its own 1:1 reference line.

S.4.2.2 Comparison with analyses of datasets of 10000 SNPs

		Relative bias			Relative RMSE			coverage				
parameter	SNPs	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI	HPD CI
$\log(N_2)$	10000	**0.00268**	0.0936	0.0979	**0.0329**	0.0953	0.0997	0.98	0.965	**0.96**	0.975	1
—	5000	**0.00736**	0.101	0.106	**0.0478**	0.104	0.109	0.955	**0.948**	0.948	0.952	0.98
t_1	10000	**0.0613**	0.324	0.322	**0.24**	0.34	0.356	0.995	**0.98**	0.98	1	1
—	5000	**0.101**	0.305	0.294	0.318	0.313	**0.309**	0.985	0.968	**0.965**	1	0.998
$\log(t_{12})$	10000	**-0.00565**	-0.034	-0.0248	**0.032**	0.046	0.0403	0.975	0.97	**0.965**	1	1
—	5000	**-0.00482**	-0.0338	-0.0298	**0.0463**	0.0563	0.0578	0.96	**0.942**	0.928	1	1
$\log(1+t_{23})$	10000	**0.0278**	0.239	0.275	**0.0537**	0.24	0.275	0.975	0.96	**0.945**	0.665	0.52
—	5000	**0.0392**	0.246	0.276	**0.07**	0.246	0.276	0.972	0.97	**0.95**	0.152	0.195
N_{bn34}	10000	**0.00376**	0.352	0.312	**0.00933**	0.355	0.321	0.98	0.96	**0.955**	0.92	0.975
—	5000	**0.00764**	0.36	0.326	**0.0194**	0.363	0.334	0.992	0.98	0.982	0.885	**0.955**
$\log(1+t_{34})$	10000	**0.00391**	-0.0349	-0.00871	0.019	0.0412	**0.0168**	0.975	0.965	**0.96**	1	1
—	5000	0.00694	-0.0373	**-0.00496**	0.028	0.0468	**0.0229**	**0.948**	0.938	0.94	1	1
$\log(N_{\mathrm{a}})$	10000	**0.000985**	0.0656	0.0893	**0.0359**	0.0696	0.0933	0.98	0.965	**0.96**	1	1
—	5000	**0.00463**	0.0806	0.104	**0.0526**	0.0854	0.109	0.962	0.945	**0.948**	1	1

Table S.9: Performance summaries for the 7-parameter Human admixture scenario
Alternating rows show results for datasets of 10000 SNPs (200 simulated datasets) compared to results from Table S.4 for datasets of 5000 SNP (400 simulated datasets).

S.4.2.3 Performance when MAFs are used

	"PSM19" controls			alternative controls		
parameter	rel. bias	rel. RMSE	profLR	rel. bias	rel. RMSE	profLR
$\log(N_2)$	0.0702	0.176	0.955	0.0197	0.0746	0.98
t_1	0.172	0.359	0.96	0.136	0.342	0.975
$\log(t_{12})$	0.0245	0.12	0.965	0.000151	0.051	0.985
$\log(1+t_{23})$	0.0846	0.118	0.95	0.0533	0.0755	0.98
N_{bn34}	0.0178	0.0309	0.965	0.00997	0.019	0.995
$\log(1+t_{34})$	0.0241	0.0618	0.965	0.00832	0.033	0.99
$\log(N_{\text{a}})$	0.0411	0.129	0.965	0.0156	0.0598	0.99

Table S.10: Performance of inferences using MAFs.
Likelihood surfaces have been re-inferred for the first 200 simulated datasets and their associated final reference tables of 20000 samples. Bias and RMSE are relative to the explored parameter ranges, as in other Tables.

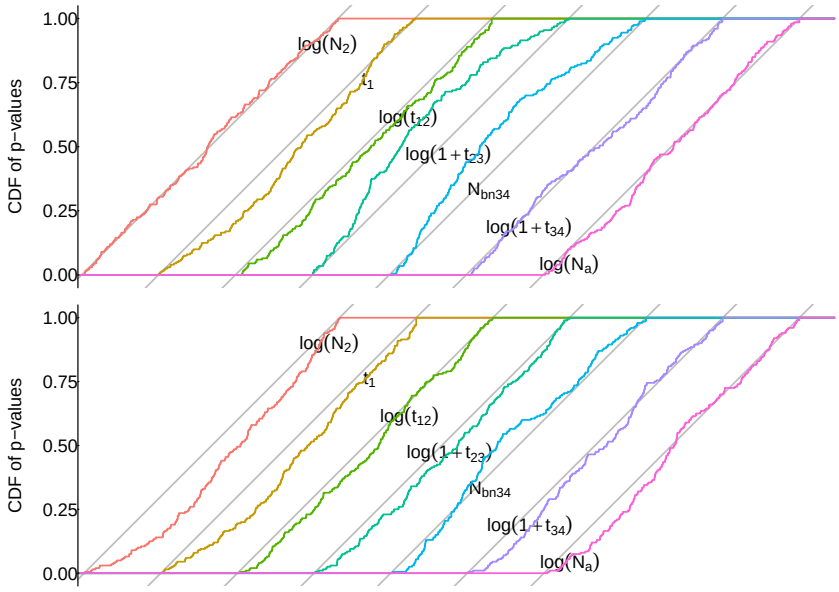

Figure S.5: Distributions of p-values of summary-LRTs using MAFs.
Top: with "PSM19" control values (see Section S.1.7); bottom: with alternative control values. Simulation details as for Table S.10.

S.4.2.4 Results with synthetic reference table

Here we have taken 100 samples at random from each of the 20000-sample reference tables (containing raw summary statistics) for each of the first 200 simulated datasets, to construct a new synthetic reference table of 20000 samples. This table was used to infer new projections and likelihood surfaces for the same 200 simulated datasets. Although the biases of resulting estimators are not necessarily large, the biases and RMSEs are uniformly larger than those of the original inferences, as can be seen in Table S.11 and Fig. S.6, and coverage is far from controlled.

parameter	rel. bias	rel. RMSE	profLR	bootLR	BcorCI
$\log(N_2)$	-0.0363	0.114	0.84	**0.85**	0.835
t_1	0.526	0.616	**0.595**	0.435	0.445
$\log(t_{12})$	-0.0617	0.106	0.67	**0.675**	0.65
$\log(1+t_{23})$	0.0518	0.114	**0.88**	0.85	0.815
N_{bn34}	-0.0112	0.14	**0.955**	0.925	0.925
$\log(1+t_{34})$	0.0172	0.102	0.725	**0.77**	0.76
$\log(N_{\text{a}})$	0.0886	0.201	0.505	**0.535**	0.495

Table S.11: Performance of inferences using synthetic reference table

.

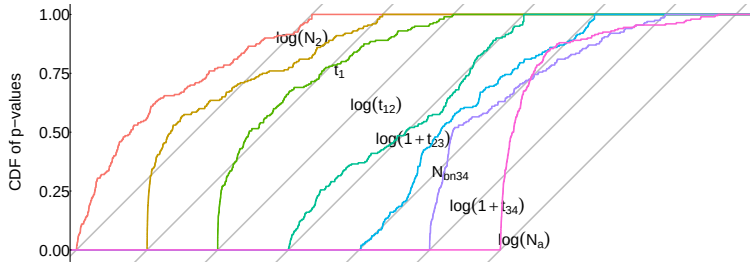

Figure S.6: Distributions of p-values of summary-LRTs using a synthetic reference table

.

S.4.2.5 Comparison with simplified sampling rule

parameter	rel. bias	rel. RMSE	coverage profLR	coverage bootLR	coverage BcorCI
$\log(N_2)$	0.00472	0.0473	**0.955**	0.955	0.942
t_1	0.114	0.33	0.97	0.962	**0.958**
$\log(t_{12})$	-0.00623	0.0462	**0.945**	0.938	0.93
$\log(1+t_{23})$	0.0329	0.0665	0.955	0.952	**0.95**
N_{bn34}	0.0068	0.019	0.992	**0.978**	0.988
$\log(1+t_{34})$	0.00654	0.0279	**0.948**	0.942	0.935
$\log(N_{\text{a}})$	0.00102	0.0523	**0.95**	0.942	0.94

Table S.12: Performance using simplified sampling rule.
These computations used the same 400 datasets as in Table 2 in the Main Text.

	default sampling rule			simplified sampling rule		
parameter	rel. bias	rel. RMSE	profLR	rel. bias	rel. RMSE	profLR
$\log(N_2)$	0.00837	0.053	0.925	0.00532	0.0553	0.925
t_1	0.134	0.379	0.972	0.135	0.38	0.968
$\log(t_{12})$	-0.00171	0.0481	0.95	-0.00376	0.0493	0.952
$\log(1+t_{23})$	0.0406	0.0771	0.91	0.042	0.0835	0.875
N_{bn34}	0.00617	0.0424	0.992	0.00705	0.0482	0.995
$\log(1+t_{34})$	0.00818	0.0306	0.93	0.00687	0.0316	0.912
$\log(N_{\text{a}})$	0.00679	0.0602	0.922	0.00387	0.0621	0.932

Table S.13: Performance of sampling rules on small reference tables.
All inferences are based on 6000-sample intermediate reference tables, built during the full 20000-sample inferences whose results are described in Tables 2 and S.12. There is only faint, if any, evidence of better performance when using the default sampling rule: marginally but consistently smaller RMSEs for all parameters; and a possibly faster convergence of coverage of profile summary-LRTs ("profLR") for $\log(1+t_{23})$, for which the importance of the iterative workflow is highlighted in the Main Text.

S.4.3 13-parameter inferences: additional simulation results

S.4.3.1 Distributions of p-values

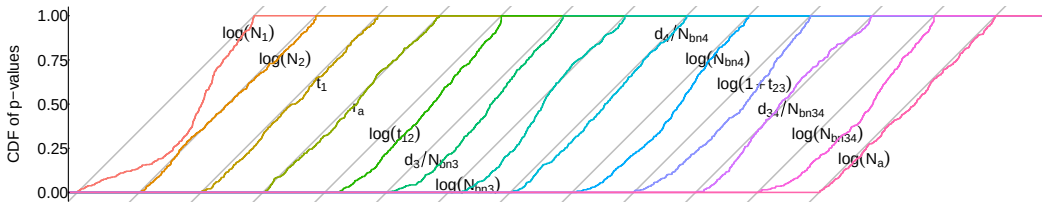

Figure S.7: Distributions of p-values of summary-LRTs for the 13-parameter human admixture scenario.

This Figure describes the same simulation results as in Table 3.

S.4.3.2 Interpretation of the compared RMSEs of summary-MLE and ABC-RF estimates

The RMSEs of summary ML estimates is lower or higher than that of ABC-RF estimates, depending on the parameter considered (Main Text Table 3). As emphasized in the Main Text, this pattern is expected and is affected by the position of data-generating values $\boldsymbol{\theta}^\dagger$ relative to prior means, so such comparisons of RMSEs are not sufficient to characterize the relative estimation performance of the two methods. The RMSEs of ABC-RF estimators are also possibly affected by other effects illustrated in the main Text. Although we did not try to tell apart these different effects for all parameters, we can look for the distinctive pattern observed in the previous simulations, and interpreted as resulting for poor exploration of parameter space by the non-iterative workflow. This pattern, where for a given parameter the ABC-RF estimates are out of the range defined by the data-generating value and the prior mean, and have a variance much lower than that of summary-MLEs, is again observed for $\log(1 + t_{23})$, but also for $\log(t_{12})$, N_{bn3}, $\log(N_{\text{bn4}})$, b_{34}, all with substantially lower variance than the SMLE [all ratios of variances < 0.28 except 0.43 for $\log(t_{12})$]. Each of these ABC-RF estimators is weakly correlated with the SMLE one for the same parameter [correlation < 0.5 except 0.57 for $\log(t_{12})$].

Although $\log(N_{\text{a}})$ appears relatively well estimated by both methods, its ABC-RF estimator also meets these diagnostic criteria, except that it has higher correlation with the SMLE (0.75).

The admixture rate r_{a}, population size N_2, and the composite parameters b_3 and b_4 appear well inferred by both methods. However, the ABC-RF estimator of $\log(N_2)$ has a suspiciously low variance ratio (0.33, versus between 0.61 and 1.31 for the three other parameters), suggesting either a strong effect of the prior mean on estimates, or an artifact of poor exploration of parameter space. The deficiency of low p-values for b_3 may result from the constraint $d_3 \geq 0$ on the parameter space and the lack of information about N_{bn3} on this boundary (See Section S.4.3.4 below for a detailed analysis).

t_1, N_1 and N_{bn34} are poorly estimated by both methods. This is less immediately obvious for N_1 and N_{bn34}, as their ABC-RF estimators have low RMSE, but in these cases the data-generating value $\theta^\dagger$ is close to the prior mean. Fig. S.8 shows that the point estimates for both $\log(N_1)$ and $\log(N_{\text{bn34}})$ remain close to their prior mean in the reference table, independently of the data-generating value, a result expected when the data provide

not information about the parameter. This also explains why the posterior estimates have much lower variance than summary-MLEs (variance ratios 0.025 and 0.039). For $\log(N_{\text{bn34}})$, this interpretation may be overlooked as the prior mean in the reference table differs from the mid-range because of the additional constraints on parameters (Main Text eq. 5).

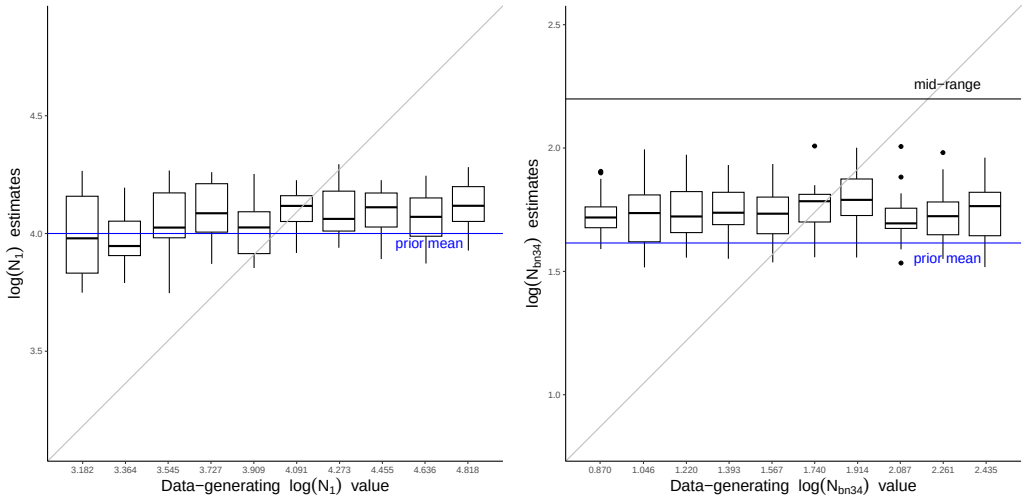

Figure S.8: Distributions of ABC-RF estimates for the parameters $\log(N_1)$ (left) and $\log(N_{\text{bn34}})$ (right).
Twenty simulated samples were analyzed for each given value of the variable parameter. Other parameters are set as in Table 3, except that in the simulations for variable N_{bn34}, the data-generating t_{23} value was increased ($\log(1 + t_{23}) = 2.5$) in order to increase the range of $\log(N_{\text{bn34}})$ values compatible with the parameter constraints for this model (Main Text eq. 5).

S.4.3.3 Reference tables of 50000 samples

parameter	rel. bias	rel. RMSE	profLR	bootLR	BcorCI
$\log(N_1)$	-0.119	0.37	0.975	**0.95**	0.95
$\log(N_2)$	-0.00705	0.033	0.97	**0.945**	0.945
t_1	0.0458	0.229	0.985	0.925	**0.95**
r_{a}	-0.00193	0.0149	0.98	0.94	**0.945**
$\log(t_{12})$	-0.0214	0.0566	1	0.97	**0.965**
d_3/N_{bn3}	0.00621	0.0327	1	**0.995**	0.995
$\log(N_{\mathrm{bn3}})$	-0.0767	0.155	0.995	**0.98**	0.99
d_4/N_{bn4}	0.00171	0.0223	0.975	**0.945**	0.955
$\log(N_{\mathrm{bn4}})$	-0.0163	0.191	0.98	0.935	**0.95**
$\log(1+t_{23})$	0.024	0.114	0.98	**0.96**	0.975
d_{34}/N_{bn34}	-0.034	0.0552	0.995	**0.985**	0.995
$\log(N_{\mathrm{bn34}})$	-0.00101	0.134	0.985	**0.96**	0.96
$\log(N_{\mathrm{a}})$	-0.0121	0.0413	0.96	**0.955**	0.96

Table S.14: Performance of summary-likelihood for the 13-parameter Human admixture scenario.
To obtain these results, the inferences initially performed on the first 200 of the 400 datasets considered in Table 3, were run for more iterations, extending the reference tables to 50,000 simulations. There is no detectable overall effect of extending the reference table on the bias and RMSE.

S.4.3.4 Analyzing the performance of summary-LRTs for the composite parameter d_3/N_{bn3}

In Main Text Table 3, d_3/N_{bn3} appears estimable with some precision. Yet the confidence intervals appear to have higher than nominal coverage, which suggests that there is little information about this composite bottleneck intensity parameter. These observations can be reconciled as follows. Short bottlenecks have little impact on genetic polymorphism. Thus, for the lowest values of bottleneck length d_3, there is little power to distinguish strong versus weak bottlenecks (corresponding to low versus high N_{bn3}). For $d_3 = 0$ in particular the log-likelihood should be a constant ℓ_0, independent of N_{bn3}. Further, for datasets whose summary-MLE for d_3 approaches the boundary, the profile likelihood for $d_3 = 0$ (or $d_3 = 1$) may not be much lower than the likelihood maximum (for simplicity of exposition, we consider the likelihood

only a function of d_3 and N_{bn3} here). These patterns are illustrated in Fig. S.9. Since the likelihood isoline for the tested value b_3^0 of $b_3 = d_3/N_{\text{bn3}}$ has a point at the boundary, the profile likelihood for b_3^0 is at least the log-likelihood ℓ_0 at the boundary. Hence, the likelihood ratio statistic for the value b_3^0 of the composite bottleneck parameter should be small, and the power of the test should be low, when the summary-MLE for bottleneck length d_3 approaches the boundary.

To check this interpretation, we performed additional simulations in which we reduced the number of estimated parameters but increased the number of simulated datasets (500, including the 200 of the original 13-parameter inferences). Only t_{12}, N_{a} and the two bottleneck parameters d_3/N_{bn3} and $\log(N_{\text{bn3}})$ were estimated, the other ones being fixed to their true values. The deficiency of low p-values is retained in these simulations (Table S.15).

parameter	rel. bias	rel. RMSE	profLR	bootLR	BcorCI
$\log(t_{12})$	0.00291	0.0208	0.966	0.96	**0.958**
d_3/N_{bn3}	0.00836	0.0534	0.996	**0.984**	0.99
$\log(N_{\text{bn3}})$	-0.075	0.193	**0.964**	0.914	0.918
$\log(N_{\text{a}})$	-0.000583	0.0153	0.93	**0.932**	0.928

Table S.15: Performance of summary-likelihood for 4-parameters inferences. Details of simulations are as for the 13-parameter inferences, except that the $\log(N_{\text{bn3}})$ range was extended to $[0, \log(50000)]$, and the inference for each dataset is based on final references tables of 11000 samples.

Since deficiencies of low p-values of the profile summary-LRT of b_3^0 are expected more for datasets whose d_3 estimates are low, we show separately in Fig. S.10 the distributions of p-values for datasets with low versus datasets with high d_3 estimates, the threshold d_3 being the data-generating value $d_3 = 32$ in these simulations. There are deficiencies in both cases, but they are, as expected, stronger for low $\hat{d}_3$. For high $\hat{d}_3$, the bootstrap test approximately corrects the distribution.

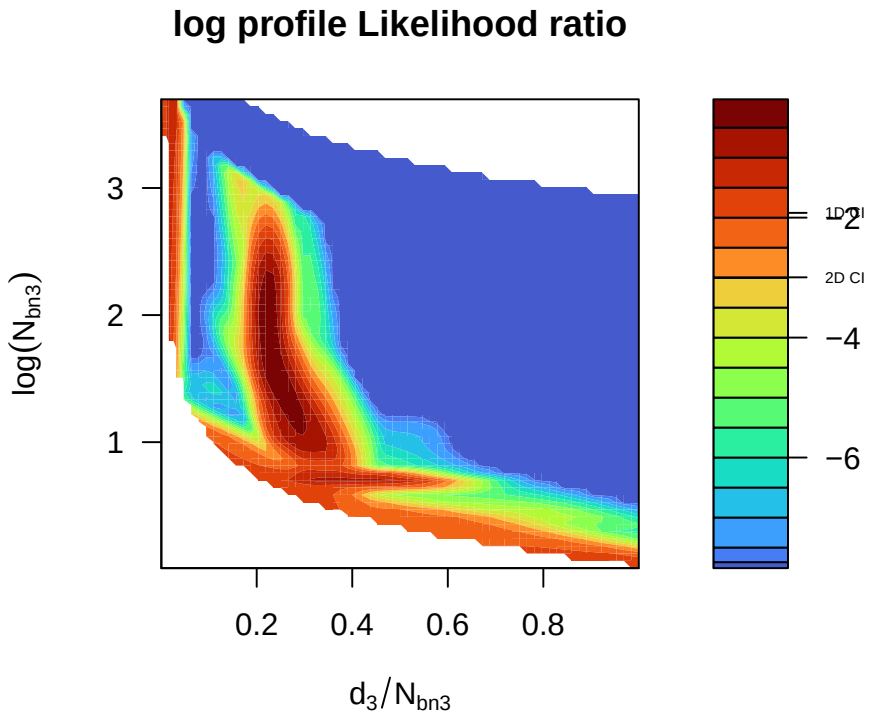

Figure S.9: Illustrative likelihood surface profile for bottleneck parameters. This Figure illustrates a recurrent pattern for different simulated datasets: the narrow vertical ridge provides the $\hat{b}_3$ estimate near the true value 0.26, but likelihoods remain relatively high along the lower boundary of the plotted area, which corresponds to vanishing bottleneck length.

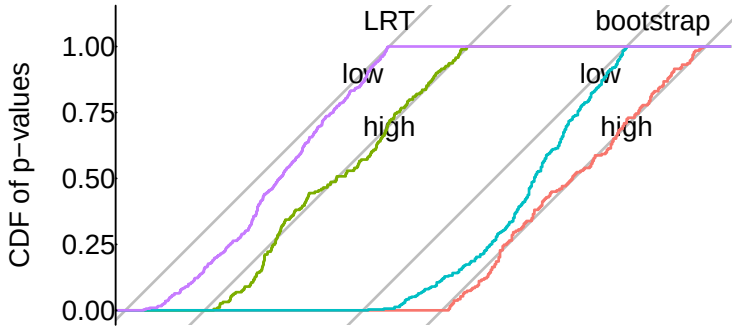

Figure S.10: Distributions of p-values of tests for the composite parameter d_3/N_{bn3}.
Distributions are shown for likelihood ratio (left) and bootstrap tests (right), distinguishing in each case the distributions for datasets with low estimates of bottleneck length, $\hat{d}_3 < 32$, from those with higher estimates (189 and 311 datasets, respectively).

S.4.3.5 Inference without composite parameters

Results for the 13-parameter variant without composite bottleneck intensity parameters are detailed in Table S.16 and Fig. S.11. Only the admixture parameter r_{a} can be estimated precisely, with approximately uniform distribution of p-values. d_4 comes second in terms of RMSE. At the opposite, MSLE estimators of t_1, t_{12}, and the population size parameters all have high relative RMSEs (from 0.195 to 0.528), and large deficiencies of low p-values. These results confirm the identifiability issues that led us to consider the alternative 13-parameter variant with composite bottleneck intensity parameters. By applying the same diagnostic criteria as in Section S.4.3.2, one can see that the ABC-RF results for $\log(N_1)$, $\log(N_3)$, $\log(1+t_{23})$ and $\log(1+t_{23})$ are suspect because the posterior-mean estimator falls on average out of the interval defined by the prior mean and the data-generating parameter value. All ABC-RF estimators except the one for r_{a} are suspect also in terms of their low variance relative to summary-MLEs: the variance ratios are 0.964 for r_{a}, but 0.216 for d_4, 0.14 for $\log(N_2)$, 0.13 for d_{34}, and < 0.06 for all other parameters.

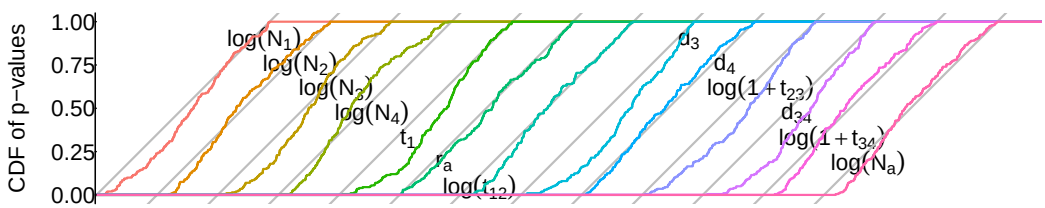

Figure S.11: Distributions of p-values of summary-LRTs for the 13-parameter Human admixture scenario without composite parameters.

	Relative bias			Relative RMSE			coverage				
parameter	MSLE	postEv	postMed	MSLE	postEv	postMed	profLR	bootLR	BcorCI	postCI	HPD CI
$\log(N_1)$	**0.374**	0.465	0.485	0.528	**0.469**	0.492	0.995	0.965	0.98	**0.955**	0.93
$\log(N_2)$	0.152	0.142	**0.114**	0.2	0.15	**0.125**	1	**0.99**	0.995	1	1
$\log(N_3)$	0.293	0.131	**0.122**	0.374	**0.143**	0.144	0.995	**0.985**	0.995	1	1
$\log(N_4)$	0.325	0.253	**0.249**	0.407	0.257	**0.256**	0.995	**0.985**	0.99	1	1
t_1	0.264	0.251	**0.203**	0.431	0.256	**0.217**	0.995	**0.985**	0.985	1	1
r_{a}	**-0.000951**	0.00831	0.00879	**0.0155**	0.0173	0.0183	0.96	0.96	**0.95**	1	1
$\log(t_{12})$	0.138	0.0569	**0.0324**	0.195	0.0595	**0.0382**	1	**0.995**	0.995	1	1
d_3	**0.00966**	-0.126	-0.0997	0.104	0.128	**0.102**	0.99	**0.985**	0.99	1	1
d_4	**0.035**	0.0458	0.0472	0.0812	**0.0571**	0.0618	0.985	**0.975**	0.98	1	1
$\log(1+t_{23})$	**0.0637**	0.114	0.122	**0.1**	0.114	0.123	0.985	**0.98**	0.99	1	1
d_{34}	**-0.0636**	-0.097	-0.0898	0.12	0.104	**0.0967**	0.995	0.995	**0.99**	1	1
$\log(1+t_{34})$	0.0862	**-0.00234**	0.0361	0.11	**0.0166**	0.0397	1	**0.995**	0.995	1	1
$\log(N_{\mathrm{a}})$	0.149	0.0411	**0.0318**	0.196	0.0454	**0.0399**	1	**0.99**	0.99	1	1

Table S.16: Performance for the 13-parameter Human admixture scenario without composite bottleneck parameters.